agronomy

MDPI

Article

A Two-Layer Signal Validation and Temporal Analysis Framework for Environmental Sensor Fault Detection in a Small-Scale Vertical Farm

Md Ashikur Rahman [1], Md Nasim Reza [1,2], Ezatullah Zakir [2], Ji-Yeon Kang [3], Md Mamunur Rashid [1], Sakib Robin [1] and Sun-Ok Chung [1,2,*]

[1] Department of Smart Agricultural Systems, Graduate School, Chungnam National University, Daejeon 34134, Republic of Korea; rahmanashikur@o.cnu.ac.kr (M.A.R.); reza5575@cnu.ac.kr (M.N.R.); mamun@o.cnu.ac.kr (M.M.R.); s.robin97@o.cnu.ac.kr (S.R.)
[2] Department of Agricultural Machinery Engineering, Graduate School, Chungnam National University, Daejeon 34134, Republic of Korea; zak14@o.cnu.ac.kr
[3] Bonghwa Insect Hotel Agricultural Corporation, Ltd., Bonghwa 36218, Republic of Korea; insecthotel0941@gmail.com
* Correspondence: sochung@cnu.ac.kr; Tel.: +82-42-821-6712

Abstract

Reliable environmental sensing is essential for stable monitoring and control in vertical farms; however, high data-acquisition completeness does not ensure that measurements are valid, temporally consistent, or operationally meaningful. This study proposed an interpretable two-layer framework that distinguished direct signal-quality errors from in-range temporal and operating-context abnormalities in a Wi-Fi-based small-scale vertical farm. Nine sensor nodes were installed across nine cultivation beds under empty-bed and continuous LED-on conditions. Air temperature, relative humidity, CO_2 concentration, and light intensity were measured at 30 s intervals for 10 days and transmitted through MQTT, Node-RED, and InfluxDB on a Raspberry Pi local server. Layer 1 screened timestamp discontinuities, invalid or null packets, and physically invalid measurements. Layer 2 evaluated only valid records using moving-window variability, rate of change, peer-node residuals, and LED-on context consistency, and a study-specific F-code taxonomy converted detected abnormalities into interpretable operating-status labels. Detection thresholds were determined from a separate two-day period immediately preceding the 10-day evaluation and were not adjusted using the evaluation records. Of 259,200 expected records, 258,910 were received (99.888% completeness). Layer 1 identified 290 missing timestamp positions, 25 invalid packets, and 40 out-of-range records. The framework achieved 99.68% accuracy, 98.49% precision, 98.89% recall, 98.69% binary F1 score, and 84.69% macro-averaged F-code F1 score. Combining signal validation with temporal and context analysis detected abnormalities that static range validation could not identify.

Keywords: vertical farm; environmental sensor; sensor fault detection; signal validation; data quality; operational status monitoring

Check for updates

Academic Editor: Jorge Antonio Sánchez-Molina

Received: 8 September 2026
Revised: 8 October 2026
Accepted: 14 October 2026
Published: 16 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Vertical farming has become an important controlled-environment agricultural production system for improving crop productivity, land use efficiency, and resource use efficiency under indoor growing conditions. In plant factory and vertical farm systems, crops are cultivated in vertically stacked layers where environmental regulation is achieved

through artificial lighting, ventilation, temperature control, humidity control, and carbon dioxide management [1–5]. The stability of air temperature, relative humidity, carbon dioxide concentration, and light intensity is critical because these variables influence plant growth, photosynthesis, transpiration, gas exchange, and energy use [1,3,6]. Therefore, reliable environmental sensing is required to support closed-loop environmental regulation, stable crop production, and operational decision-making in vertical farms [6,7].

Recent developments in smart agriculture have increased the use of Internet of Things-based monitoring platforms, wireless sensor nodes, local servers, time-series databases, and real-time dashboards in controlled-environment agriculture [6,7]. These systems allow environmental data to be collected from multiple cultivation beds, rack layers, or production zones and can support historical analysis, alarm generation, and automated control decisions [6–9]. In multi-tier vertical farms, environmental gradients may occur because of differences in rack height, airflow distribution, crop canopy position, LED distance, and local ventilation conditions [3,4,6]. Therefore, multi-node sensing is necessary to capture spatial variability, vertical variability, and temporal variability across cultivation positions rather than relying on a single environmental sensor for the whole growing system [6,7].

However, environmental data collected from in situ sensors are not always reliable. Sensor streams may include missing records, communication timeouts, packet loss, invalid values, out-of-range measurements, random noise, abrupt spikes, stuck responses, bias, and drift [10–15]. These problems may be caused by hardware aging, calibration loss, condensation, dust accumulation, unstable power supply, electrical interference, poor sensor placement, cable or connector instability, and wireless communication interruption [12,13,16]. In enclosed crop environments, sensor faults are particularly problematic because abnormal sensor outputs may be confused with genuine spatial, vertical, or temporal environmental variability. For example, a drifting carbon dioxide sensor may appear to indicate gradual ventilation change, while a stuck temperature signal may appear to represent stable canopy-level conditions [16–18].

Traditional sensor fault-detection approaches include hardware redundancy, statistical process control, and threshold-based range validation [19,20]. Hardware redundancy can improve reliability, but it increases installation cost, maintenance effort, and wiring complexity, which may be impractical for small-scale vertical farm research platforms. Statistical process control can detect large deviations from baseline behavior, but its performance depends strongly on stable baseline assumptions and suitable control limits [20]. Simple threshold-based validation can detect physically impossible readings, such as relative humidity above 100%, negative carbon dioxide concentration, or negative light intensity [12,13]. However, fixed numerical bounds cannot reliably detect in-range anomalies such as a plausible but stuck temperature value, gradual sensor drift, excessive short-term fluctuation, or a low light reading during an LED-on period [16–18].

Advanced computational approaches, including model-based diagnosis, machine learning, and deep learning, have also been applied to anomaly detection and sensor abnormality detection and operating-status classification [16–19]. These methods can provide strong detection performance when sufficient training data, computational resources, and representative fault labels are available [16–19]. Nevertheless, their use in small-scale vertical farm systems can be limited by the need for large labeled datasets, high computational burden, and reduced interpretability of detected abnormal conditions [16–18]. In practical farm operations, the detection result should not only indicate whether an abnormal event occurred but should also support maintenance decisions by identifying likely fault types such as communication failure, out-of-range measurement, stuck response, unstable signal, transient spike, drift-like behavior, or context inconsistency [16,19].

Previous studies on sensor data quality, time-series anomaly detection, and agricultural IoT sensor fault diagnosis have shown that sensor reliability depends on more than measurement accuracy alone [10–13,16–19,21–23]. Recent controlled-environment and smart-greenhouse studies have further demonstrated the use of machine-learning-based, AI-based, residual-based, and signal-processing-based approaches for detecting abnormal sensor or system behavior [24–28]. These studies confirm the importance of detecting abnormal sensor behavior before measurements are used for environmental monitoring or control; however, a practical distinction remains between faults that can be identified directly from data availability or physical validity and abnormalities that remain numerically plausible but become evident only through temporal or operating-context analysis. Furthermore, highly model-dependent approaches may require representative training datasets or detailed process models, whereas small experimental vertical-farm systems may benefit from transparent rules that can be inspected and modified by operators. Therefore, the contribution of the present study is not sensor fault detection alone, but the integration of communication-quality screening, physical signal validation, temporal characterization, and operating-context interpretation within a sequential and maintenance-oriented framework. Compared with existing statistical-threshold, residual-based, time-series anomaly-detection, and smart-greenhouse fault-detection approaches, the proposed framework integrates communication-quality screening, physical signal validation, temporal analysis, peer-node comparison, and operating-context interpretation within a sequential two-layer structure. This design emphasizes transparent, rule-based detection of both direct data-quality errors and in-range temporal abnormalities for local vertical-farm monitoring.

Accordingly, this study developed and evaluated an interpretable two-layer framework for environmental sensor fault detection in a Wi-Fi-based vertical-farm monitoring system. The specific objectives were to (i) characterize calibration performance and data-acquisition reliability across nine environmental sensor nodes; (ii) develop a first-layer procedure for identifying communication failures, invalid packets, missing values, and physically invalid measurements; (iii) develop a second-layer procedure for detecting in-range temporal abnormalities, including stuck responses, excessive variability, transient spikes, drift-like deviations, and operating-context inconsistencies; (iv) organize the detected conditions using a study-specific F-code operating-status taxonomy; and (v) evaluate the framework using binary and F-code-level detection metrics, threshold-robustness analysis, and local-server computational performance.

2. Materials and Methods

2.1. Experimental Small-Scale Vertical Farm

The experiment was conducted in Section 4 of an indoor vertical farm facility at Chungnam National University, Daejeon, Republic of Korea. The facility had an approximate floor area of 7.0 m × 3.0 m and was divided into multiple cultivation sections. Section 4 was selected as the experimental monitoring area because it contained a three-layer cultivation rack suitable for bed-level environmental sensor placement.

The experimental rack consisted of three vertical cultivation layers and nine cultivation beds. One environmental sensor node was installed in each cultivation bed, resulting in a total of nine monitoring nodes. The rack length and width were approximately 2.70 m and 0.60 m, respectively, and the front width of the monitored rack section was approximately 2.80 m. Each sensor node was positioned near the central sensing area of the assigned cultivation bed. Based on the rack schematic, the sensor-node position was located approximately 25 cm from the upper and lower reference positions within each bed area.

The experiment was conducted under empty-bed conditions, and no crop was grown during the monitoring period. Therefore, crop species, growth stage, days after transplanting, and planting density were not applicable. This setup was used because the objective of this study was to evaluate environmental sensor-data reliability and fault-detection performance, not crop-growth response. The monitored variables were air temperature, relative humidity, CO_2 concentration, and light intensity.

The LED system was operated continuously during the monitoring period; therefore, all light measurements were interpreted under a continuous LED-on operating condition. Ventilation fans installed in the cultivation structure were also operated continuously. No CO_2 enrichment was applied. Room-level air conditioning was not separately controlled or logged by the sensor-node experiment and was treated as part of the background indoor environmental condition. Under this operating condition, abnormally low light intensity was interpreted as a possible context inconsistency under continuous LED-on operation.

The sensor nodes transmitted environmental measurements to the local monitoring system through Wi-Fi at a sampling interval of 30 s. The experimental layout, Section 4 rack structure, sensor-node placement, and monitoring environment are shown in Figure 1.

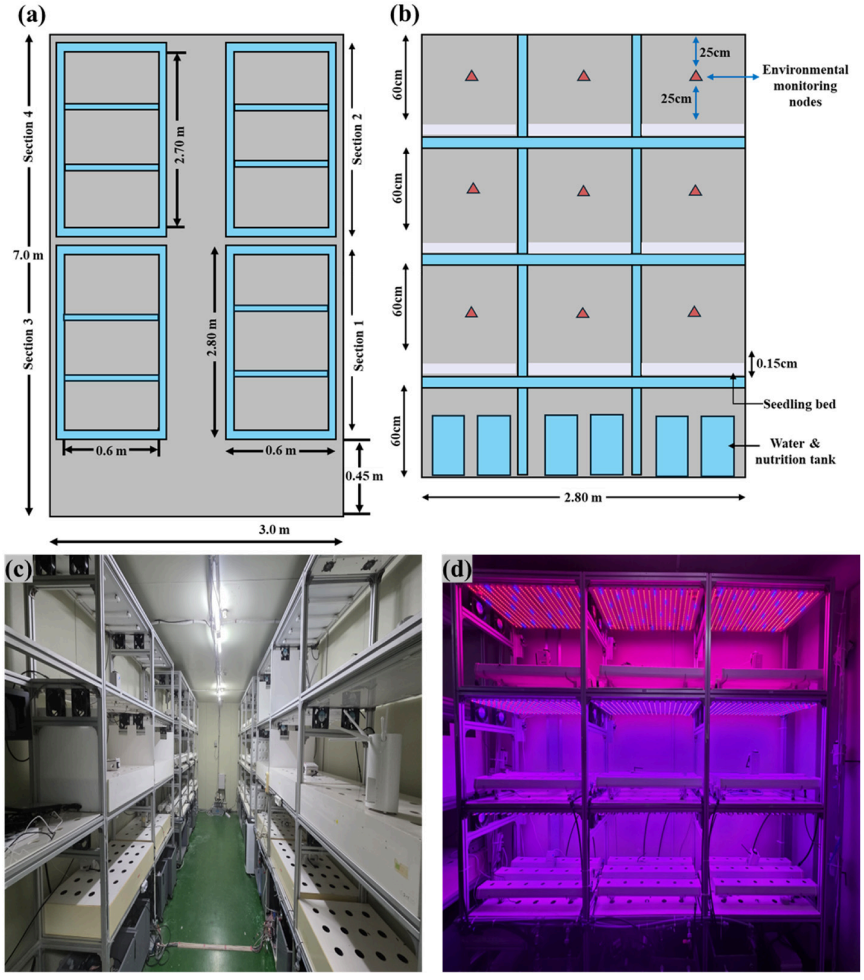


Figure 1. Experimental vertical farm facility and sensor-node placement used for environmental monitoring. (**a**) Floor layout of the indoor vertical farm facility showing the location of Section 4 used for the experiment. (**b**) Front-view schematic of the three-layer cultivation rack in Section 4 showing the nine sensor-node positions. (**c**) Photograph of the indoor vertical farm facility and monitoring aisle. (**d**) Photograph of the Section 4 rack under continuous LED-on operation. One environmental sensor node was installed in each cultivation bed to measure air temperature, relative humidity, CO_2 concentration, and light intensity at 30 s intervals.

2.2. Sensor-Node Hardware Architecture

Each environmental monitoring node was developed using an Arduino UNO R4 WiFi microcontroller board (Arduino S.r.l., Monza, Italy). A total of nine identical sensor nodes were prepared for the experiment, and node identifiers were assigned from node_01 to node_09. Each node was installed in one cultivation bed of the experimental rack and was connected to the environmental sensors required for measuring air temperature, relative humidity, CO_2 concentration, and light intensity.

The node hardware was assembled inside a protective plastic enclosure to improve wiring stability and to simplify installation in the vertical farm structure. Fixed male M12 connectors were mounted on the enclosure wall, and detachable female M12 sensor cables were connected from the sensor side. This connector-based structure allowed the external sensors to be connected, removed, or replaced without opening the enclosure. It also reduced wiring disturbance during installation, inspection, and maintenance.

The internal node architecture consisted of the microcontroller board, sensor-interface wiring, connector terminals, and power input. The temperature and relative-humidity sensor were connected to the microcontroller through a digital signal interface. The CO_2 sensor was connected through a UART interface using TX, RX, power, and ground lines. The light sensor was connected through the I^2C interface using SDA, SCL, power, and ground lines. Each node was powered using a USB adapter, and the same hardware configuration was applied to all nine nodes.

The Arduino firmware performed sensor reading, node identification, packet formatting, and wireless data transmission. The node-side program was kept lightweight so that all nodes could collect data using the same 30 s sampling interval. Preprocessing, signal validation, temporal feature calculation, and fault-code classification were not performed on the Arduino board. These processes were performed after data reception and storage on the local monitoring server.

A Raspberry Pi 4B with 8 GB RAM (Raspberry Pi Ltd., Cambridge, UK) was used as the local monitoring server. In this study, the server was used for data reception, storage, preprocessing, temporal analysis, and fault-code classification. This separation between node-side acquisition and server-side processing reduced the computational burden on the sensor nodes and supported consistent data collection from the nine-node monitoring system.

The sensor-node hardware architecture and enclosure design are shown in Figure 2. Figure 2a shows the microcontroller-based node architecture, connector interface, USB power input, and wireless transmission function. Figure 2b shows the sensor connections and Wi-Fi communication scheme, Figure 2c shows the top view of the prototype enclosure, and Figure 2d shows the assembled enclosure used for detachable sensor connection. The main hardware components and their functions are summarized in Table 1.

Table 1. Hardware components and functions of the environmental monitoring node and local server.

Component	Model or Type	Main Function	Number Used
Sensor-node microcontroller	Arduino UNO R4 Wi-Fi	Sensor reading, node identification, packet formatting, and wireless data transmission	9
Node enclosure	Plastic protective enclosure	Protection of microcontroller, wiring, and connector terminals	9
Sensor connector	Fixed male M12 connector with detachable female sensor cable	Detachable external sensor connection and maintenance	9 sets

Table 1. *Cont.*

Component	Model or Type	Main Function	Number Used
Temperature/RH sensor interface	Digital signal interface	Air temperature and relative-humidity sensor connection	9
CO_2 sensor interface	UART interface	CO_2 sensor communication using TX, RX, power, and ground lines	9
Light sensor interface	I^2C interface	Light sensor communication using SDA, SCL, power, and ground lines	9
Node power supply	USB adapter	Power supply for each sensor node	9
Local monitoring server	Raspberry Pi 4B, 8 GB RAM	Data reception, storage, preprocessing, temporal analysis, and fault-code classification	1

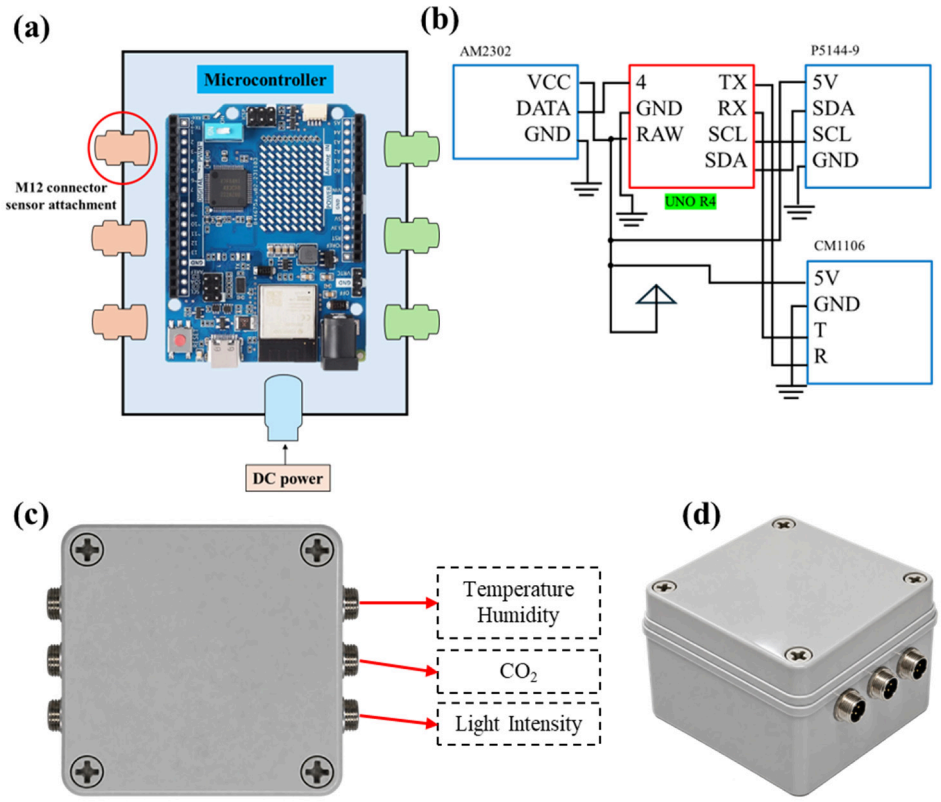


Figure 2. Sensor-node hardware architecture and enclosure design used for vertical farm environmental monitoring. (**a**) Sensor-node hardware schematic showing the Arduino UNO R4 Wi-Fi microcontroller, sensor-interface ports, and USB adapter power input. (**b**) Sensor connection and Wi-Fi communication scheme showing the connection between external environmental sensors and wireless data transmission from the sensor node. (**c**) Top view of the prototype enclosure with fixed M12 connector ports mounted on the enclosure wall. (**d**) Side view of the assembled sensor-node enclosure used for detachable external sensor connection and stable installation in the vertical farm rack.

2.3. Environmental Sensors and Measurement Variables

The environmental monitoring system measured four variables: air temperature, relative humidity, CO_2 concentration, and light intensity. These variables were selected because they represent the main indoor environmental conditions required for vertical farm monitoring. Since the experiment was conducted under empty-bed conditions, the measured

variables were used to evaluate sensor-data reliability and fault-detection performance rather than crop-growth response.

Air temperature and relative humidity were measured using a digital temperature–humidity sensor (Model: DHT22, Aosong Electronics Co., Ltd., Guangzhou, China). These two variables were evaluated together because they describe the thermal and moisture conditions inside the cultivation bed environment. Under stable indoor operation, temperature and relative humidity were expected to change gradually; therefore, abrupt changes, physically invalid values, excessive short-term fluctuation, and constant-output behavior were considered important signal patterns for fault detection. CO_2 concentration was measured using a CO_2 sensor (model: CM1106, Cubic Sensor and Instrument Co., Ltd., Wuhan, China). CO_2 was included because it can show stronger temporal variation than temperature and relative humidity due to ventilation, indoor air mixing, and human activity near the experimental facility. Therefore, CO_2 measurements were evaluated using both physical range validation and temporal indicators, including rate of change, moving-window variability, stuck-response behavior, and peer-residual drift-like deviation.

Light intensity was measured as illuminance in lux using the digital light sensor module (model: P5144-9, Hanjin Data, Incheon, Republic of Korea). The sensor output was calibrated against a digital lux meter (model: AR823+, Dongguan Wanchuang Electronic Products Co., Ltd., Dongguan, China) under controlled lighting conditions. Because the reference instrument measured illuminance in lux and the spectral distribution of the horticultural LED system was not independently measured, light intensity was analyzed and reported only in lux. No conversion from lux to PPFD was applied in the final analysis. The LED system was operated continuously during the monitoring period; therefore, persistently low light intensity under this continuous LED-on condition was treated as a context-inconsistency signal because it could indicate sensor blockage, sensor displacement, local shading, sensor malfunction, or lighting-system abnormality. The same sensor configuration was used for all nine monitoring nodes, and all sensors were installed outside the enclosure through detachable M12 sensor cables. The specifications of the environmental sensors used in the nine-node monitoring system are summarized in Table 2.

Table 2. Environmental sensor specifications used in the nine-node monitoring system.

Variable	Sensor Model	Manufacturer	Measurement Range	Accuracy	Interface
Air temperature	DHT22	Aosong Electronics Co., Ltd., Guangzhou, China	−40 to 80 °C	±0.5 °C	Digital
Relative humidity	DHT22	Aosong Electronics Co., Ltd., Guangzhou, China	0 to 100% RH	±2% RH	Digital
CO_2 concentration	CM1106	Cubic Sensor and Instrument Co., Ltd., Wuhan, China	0 to 5000 ppm	±50 ppm	UART
Light intensity	P5144-9	Hanjin Data, Incheon, Republic of Korea	1 to 65,535 lx	±20% lx	I^2C

2.4. Wi-Fi Communication and Data Logging

Environmental sensor data were transmitted from the nine sensor nodes to a local monitoring server through a Wi-Fi-based communication network. The communication system consisted of Arduino UNO R4 WiFi sensor nodes, an ipTIME N804R Wi-Fi router

(EFM Networks Co., Ltd., Yongin, Republic of Korea), an MQTT broker, Node-RED, InfluxDB, and a server-side fault-detection analysis module. This architecture separated sensor-side data acquisition from server-side data reception, storage, preprocessing, and fault-detection analysis.

Each sensor node measured air temperature, relative humidity, CO_2 concentration, and light intensity at 30 s intervals. After each sensing cycle, the Arduino firmware formatted the measurements as a JSON payload containing the node identifier and sensor values. The node identifiers were assigned from node_01 to node_09. The payload was transmitted through the ipTIME N804R router using a 2.4 GHz IEEE 802.11 b/g/n wireless network. The data packet structure and stored variables are summarized in Table 3.

Table 3. Data packet structure and stored variables used for environmental monitoring.

Field Name	Description	Unit or Format
timestamp	Server-side timestamp assigned during data reception and storage	YYYY-MM-DD HH:MM:SS
node_id	Sensor node identification number	node_01 to node_09
bed_id	Cultivation bed identification number	bed_01 to bed_09
temperature	Air temperature	°C
relative_humidity	Relative humidity	%RH
CO_2	CO_2 concentration	ppm
light intensity	Light intensity (illuminance)	lx
packet_status	Optional communication or validity status	valid/missing/invalid
server_time	Optional server-side reception time	timestamp

Sensor data were published to an MQTT broker installed on the Raspberry Pi 4B local server. Eclipse Mosquitto v2.1.2 was used as the MQTT broker and was operated through port 1883. The MQTT topic used for environmental sensor-data transmission was farm/sensor/data. MQTT messages were transmitted using QoS 0 because the nodes generated updated environmental measurements every 30 s and the analysis focused on continuous time-series behavior rather than guaranteed delivery of every individual packet. When Wi-Fi or MQTT communication was interrupted, the node firmware attempted reconnection before publishing subsequent measurements. Node-RED v4.1.10, operated with Node.js v20.20.2, was used as the data reception and processing layer. Node-RED subscribed to the MQTT topic, received the incoming JSON payload, parsed the sensor fields, assigned node metadata, and prepared the records for time-series storage. The parsed records were stored in InfluxDB v2.9.1 using the organization PA Lab and the bucket sensor_data. The stored time-series dataset was then used for preprocessing, signal validation, temporal analysis, and fault-detection evaluation.

Timestamps were assigned at the local monitoring server during data reception and storage. This server-side timestamping method provided a common time base for all stored records because records from all nodes were indexed using the Raspberry Pi system clock. Therefore, timestamp-continuity analysis and missing-update detection were performed using the server-side stored timestamps. A missing update was identified when the time gap between two consecutive records from the same node exceeded 60 s, corresponding to twice the programmed 30 s sampling interval.

Because independent synchronization between node-side and server-side clocks was not documented, node-to-server reception delay was not treated as an absolute communication-latency metric in this study. Instead, communication reliability was evaluated using expected packet count, received packet count, timestamp continuity, invalid-packet status, and missing-update events. This avoided overinterpreting timing differences that could be affected by unsynchronized clocks.

The complete data communication and logging workflow is illustrated in Figure 3. Sensor measurements were acquired by the Arduino-based nodes and transmitted through the Wi-Fi router to the MQTT broker on the local server. Node-RED received and parsed the MQTT payloads, and InfluxDB stored the processed time-series records. The stored data were subsequently used for preprocessing, signal validation, temporal analysis, and fault-detection evaluation.

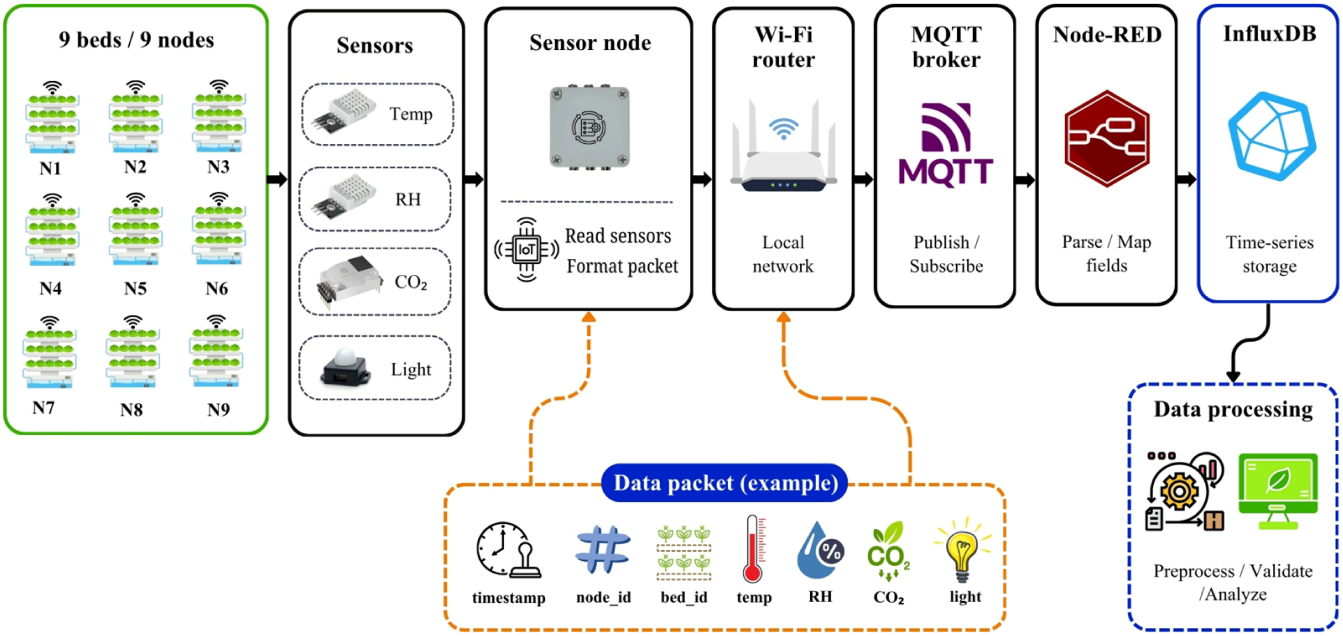


Figure 3. Wi-Fi-based data communication and logging workflow used for environmental sensor monitoring.

The environmental sensor nodes acquired air temperature, relative humidity, CO_2 concentration, and light intensity at 30 s intervals. The data packets were transmitted through the Wi-Fi router to the MQTT broker; received and parsed in Node-RED; stored in the InfluxDB time-series database; and used for preprocessing, signal validation, temporal analysis, and fault-detection evaluation. The implemented Node-RED data-processing flow and InfluxDB time-series storage interface are shown in Figure 4.

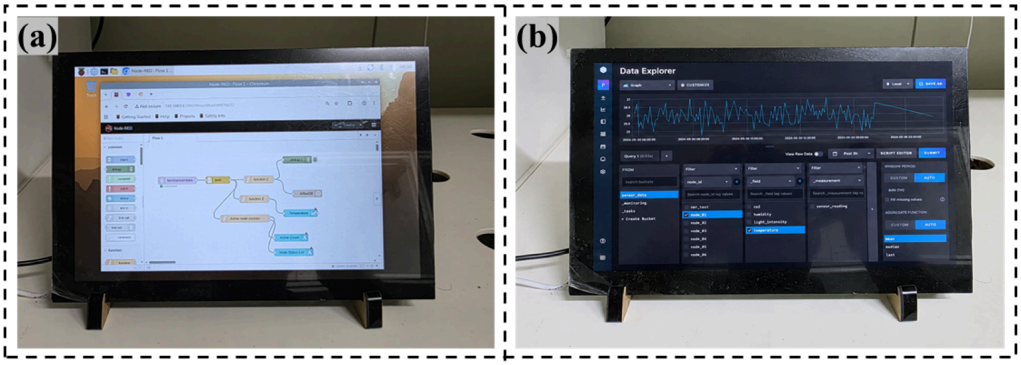


Figure 4. Local server implementation on the Raspberry Pi 4B. (**a**) Node-RED flow used for MQTT subscription, JSON parsing, and field mapping and (**b**) InfluxDB Data Explorer showing stored node-wise time-series records in the sensor_data bucket.

2.5. Sensor Calibration and Reference Measurement

Sensor calibration was performed before the monitoring experiment to reduce systematic measurement bias in the environmental sensing configuration. One representative

sensor node was calibrated under controlled conditions because the nine deployed nodes used the same sensor models, connector structure, wiring configuration, firmware, sampling interval, and data-transmission procedure. The calibration equations obtained from the representative node were applied as common sensor-type correction models for the nine-node monitoring system. This approach was intended to provide configuration-level correction rather than individual calibration of each sensor unit. Possible unit-to-unit variation among sensors of the same model was therefore not eliminated by this procedure. Independent reference calibration or validation measurements were not available for the remaining eight deployed nodes; therefore, the common correction model was not independently verified for each individual sensor unit. Therefore, the calibration results represented sensor-configuration correction performance and were not interpreted as node-specific calibration performance for all deployed nodes.

A CO_2 growth chamber was used for the calibration of the temperature, relative humidity, and CO_2 sensors. The chamber controller was used to set and monitor the calibration conditions. The chamber was not treated as an independently traceable reference instrument beyond the stated controller specifications. The representative sensor node was placed near the chamber sensing location, and the chamber condition was stabilized before each calibration run. For light-intensity calibration, the light sensor was compared with a reference (SMART SENSOR AR823+) digital lux meter under controlled lighting conditions. Because the reference device measured illuminance in lux, light calibration was conducted on a lux basis. All light-intensity values were therefore calibrated, analyzed, and reported in lux, and no cross-unit conversion was applied at any stage of the analysis. The calibration setup and reference-measurement workflow are shown in Figure 5.

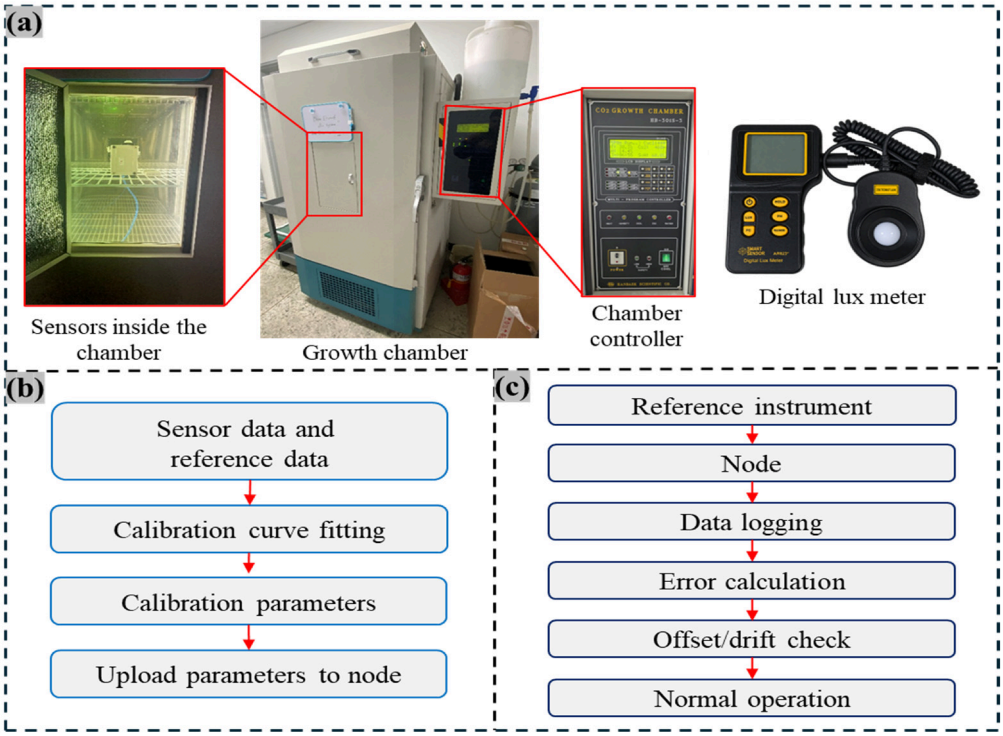


Figure 5. Calibration setup and reference-measurement workflow. (**a**) Growth chamber representative sensor node placement inside the growth chamber and chamber. (**b**) Calibration parameter generation using paired sensor and reference data. (**c**) Calibration procedure from reference measurement and sensor-node data logging to error calculation, offset and drift checking, and normal operation.

At each calibration setpoint, the raw sensor output and the corresponding reference or controlled-condition value were recorded simultaneously. Air temperature and relative

humidity were logged at 0.5 Hz for 10 min, resulting in 300 samples per setpoint. CO_2 concentration was logged at 1 Hz for 5 min, resulting in 300 samples per setpoint. Light intensity was logged at 8 Hz for 5 min, resulting in 2400 samples per setpoint. Repeated measurements collected at the same setpoint were averaged before calibration analysis, and each setpoint mean was treated as one independent observation. The calibration protocol for each environmental variable is summarized in Table 4.

Table 4. Calibration setup and measurement protocol for environmental sensors.

Variable	Reference Instrument	Reference Specification	Sampling Condition
Air temperature	HB-301S-3 CO_2 growth chamber, Hanbaek Scientific Co., Republic of Korea	Range: 0 °C to 60 °C Accuracy: ±0.2 °C	0.5 Hz for 10 min; 300 samples per setpoint
Relative humidity		Range: 35% to 95% RH Accuracy: ±3% RH	
CO_2 concentration		Range: 0 to 5000 ppm Accuracy: ±50 ppm	1 Hz for 5 min; 300 samples per setpoint
Light intensity	SMART SENSOR AR823+ digital lux meter, Dongguan Wanchuang Electronic Products Co., Ltd. China	Range: 1 to 200,000 lx Accuracy: $\leq 10,000\ lx$: $\pm(4\%\text{rdg} + 10\ \text{digits})$ $> 10,000\ lx$: $\pm(5\%\text{rdg} + 10\ \text{digits})$	8 Hz for 5 min; 2400 samples per setpoint

The calibration and validation datasets were separated at the setpoint level. Calibration setpoints were used to estimate the linear correction coefficients, whereas validation setpoints were excluded from coefficient estimation and were used only for error evaluation. Therefore, repeated observations at 0.5 Hz, 1 Hz, and 8 Hz were used to quantify setpoint stability and were not treated as independent calibration or validation samples. The calibration and independent validation setpoints used for each environmental variable are summarized in Table 5.

Table 5. Calibration and validation setpoint levels used for representative sensor-node calibration.

Variable	Calibration Setpoints Used For Model Fitting	Validation Setpoints Used Only for Error Evaluation
Air temperature	10, 15, 20, 25, and 30 °C (n = 5)	12, 18, 22, 27, and 29 °C (n = 5)
Relative humidity	40, 50, 60, 70, and 80% RH (n = 5)	45, 55, 65, and 75% RH (n = 4)
CO_2 concentration	400, 600, 800, 1000, and 1200 ppm (n = 5)	500, 700, 900, and 1100 ppm (n = 4)
Light intensity	Approximately 3704, 7407, 11,111, 14,815, 18,519, and 22,222 lx (n = 6)	Approximately 5556, 9259, 12,963, 16,667, and 20,370 lx (n = 5)

The relationship between the raw sensor output and the reference measurement was evaluated using the linear calibration model shown in Equation (1):

$$R_i = aS_{i,raw} + b \quad (1)$$

where $S_{i,raw}$ is the mean raw sensor output at calibration condition i, R_i is the corresponding reference measurement, and a and b are the slope and intercept obtained by regressing the reference measurements against the raw sensor measurements. In this study, the calibration curve was plotted with the sensor-measured value on the x-axis and the reference-measured value on the y-axis so that the fitted equation could be directly used to correct raw sensor

readings. After the calibration parameters were estimated, the corrected sensor value was calculated using Equation (2):

$$S_{i,corr} = aS_{i,raw} + b \quad (2)$$

where $S_{i,raw}$ is the raw sensor measurement, and $S_{i,corr}$ is the corrected sensor value. The same coefficients a and b estimated from Equation (1) were applied to each raw sensor reading so that the corrected value was expressed on the reference measurement scale.

Calibration performance was evaluated using R^2, mean absolute error, root mean square error, mean bias, and setpoint standard deviation. These metrics were calculated from validation setpoints after applying the calibration equation to raw sensor values. This setpoint-level evaluation prevented repeated measurements collected under the same chamber condition from being treated as independent validation observations.

2.6. Data Preprocessing and Quality Screening

Data preprocessing was performed to generate a consistent node-wise time-series dataset for signal validation and temporal analysis. The raw records stored in InfluxDB were queried using timestamp, node ID, bed ID, and sensor-field information. Each node and each measured variable were processed independently so that spatial differences among cultivation beds were not mixed with sensor-level abnormalities. In IoT-based sensor systems, data completeness, timestamp continuity, validity, and consistency are commonly used to identify missing records, communication loss, and logging inconsistency [10,12,13]. Therefore, records were first sorted chronologically for each node and checked against the expected 30 s sampling interval. Duplicate node–timestamp records were flagged during preprocessing. When duplicate packets contained identical sensor values, the first record was retained for numerical analysis, and the duplicated record was stored as a communication-quality event. When duplicate packets contained conflicting sensor values, the affected record was marked as invalid and excluded from temporal-feature calculation.

Sensor fields were converted to numerical format before analysis. Records containing null values, unreadable strings, non-numeric values, or invalid packet status were retained as Layer 1 abnormalities but excluded from Layer 2 temporal-feature calculation. This prevented corrupted or incomplete packets from influencing moving-window features such as standard deviation, signal range, rate of change, and trend slope. No interpolation was applied before fault detection because missing records, null fields, invalid packets, and communication timeouts were considered part of the sensor data-quality problem rather than gaps to be filled.

The expected number of records was calculated from the monitoring duration and sampling interval using Equation (3):

$$N_{exp} = \frac{D \times 24 \times 60 \times 60}{\Delta t} \quad (3)$$

where N_{exp} is the expected number of records per node, D is the monitoring duration in days, and Δt is the sampling interval in seconds. In this study, $D = 10$ days and $\Delta t = 30$s; therefore, each node generated 28,800 records during the monitoring period.

Missing updates were detected by comparing the time difference between two consecutive records from the same node using Equation (4):

$$\Delta T_i = t_i - t_{i-1} \quad (4)$$

where, ΔT_i is the time interval between two consecutive records, t_i is the timestamp of the current record, and t_{i-1} is the timestamp of the previous record. A missing update was assigned when $\Delta T_i > 60$ s. This threshold corresponded to twice the programmed

sampling interval and was used to distinguish short timing variation from communication-timeout events.

After timestamp and packet-quality screening, physical range validation was applied to each environmental variable. Values outside physically possible limits or sensor-specific measurement ranges were flagged as invalid. Relative humidity below 0% or above 100%, negative CO_2 concentration, and negative raw light intensity were treated as physically invalid values. Raw values were retained for identifying hardware-level or parsing-related invalid measurements, whereas calibrated values were used for temporal analysis. Small negative calibrated light intensity values caused by calibration offset were not treated as faults when the corresponding raw light measurement remained valid. Only negative raw-light measurements were treated as physically invalid values.

The final preprocessed dataset contained the timestamp, node ID, bed ID, raw sensor values, calibrated sensor values, packet-status flags, duplicate-packet flags, missing-update flags, physical-validity flags, and initial Layer 1 screening labels. Only records that passed packet-quality and physical-validity screening were used for Layer 2 temporal analysis. Records that failed these checks were retained as fault-relevant observations for communication and signal-validation analysis rather than being removed silently.

2.7. Characterization Indicators for Signal Validation and Temporal Analysis

Signal characterization indicators were calculated after preprocessing to support the two-layer fault-detection framework. The indicators were selected to evaluate data availability, physical validity, abrupt temporal change, local signal stability, stuck response, and drift-like deviation. Data availability and physical validity were used in Layer 1, whereas temporal variability, rate of change, stuck behavior, and residual-based drift tendency were used in Layer 2. These indicators were selected based on established sensor data-quality concepts, time-series anomaly-detection methods, and signal-based sensor fault-diagnosis approaches [10,12,13,16–19].

Let $x_{j,k}(t_i)$ represent the calibrated measurement of environmental variable j from sensor node k at timestamp t_i, where j indicates air temperature, relative humidity, CO_2 concentration, or light intensity, and k indicates Node_01 to Node_09. The expected sampling interval was denoted as Δt, and the moving-window interval was denoted as w. Raw values were used for physical-invalidity checking when calibration offset could affect near-zero values, whereas calibrated values were used for temporal characterization.

Data completeness was calculated to evaluate whether the expected timestamp sequence was maintained for each node and variable [10,12,13]. The completeness ratio was calculated using Equation (5):

$$C_{j,k,w} = \frac{N^{obs}_{j,k,w}}{N^{exp}_{j,k,w}} \tag{5}$$

where $C_{j,k,w}$ is the completeness ratio, $N^{obs}_{j,k,w}$ is the number of observed records, and $N^{exp}_{j,k,w}$ is the expected number of records for environmental variable j, node k, and window w. The missing-data rate was calculated using Equation (6):

$$M_{j,k,w} = \left(1 - C_{j,k,w}\right) \times 100 \tag{6}$$

where $M_{j,k,w}$ is the missing-data rate expressed as a percentage. This metric was used to quantify communication or logging loss within each time window.

Physical validity was evaluated by comparing each sensor measurement with the allowable physical or sensor-specific range [12,13]. If L_j and U_j represent the lower and upper valid limits for variable j, the out-of-range indicator was defined using Equation (7):

$$I_{j,k}^{out}(t_i) = \begin{cases} 1, & x_{j,K}(t_i) < L_j \text{ or } x_{j,K}(t_i) > U_j \\ 0, & L_j \le x_{j,K}(t_i) \le U_j \end{cases} \quad (7)$$

where $I_{j,k}^{out}$ indicates whether the measurement at timestamp t_i violates the valid range. The out-of-range rate in a moving window was calculated using Equation (8):

$$O_{j,k,w} = \frac{1}{n_w} \sum_{i=1}^{n_w} I_{j,k}^{out}(t_i) \times 100 \quad (8)$$

where $O_{j,k,w}$ is the out-of-range rate, and n_w is the number of records in window w. This metric was used to quantify the frequency of physically invalid values within each window.

Temporal stability was evaluated using moving-window variability and signal range. The moving-window standard deviation was used to detect unstable or noisy signals, whereas signal range was used to detect both excessive fluctuation and near-constant stuck behavior. Since mean and standard deviation are standard descriptive statistics, their equations were not repeated. The signal range was calculated using Equation (9):

$$R_{j,k,w} = x_{j,k,w}^{max} - x_{j,k,w}^{min} \quad (9)$$

where $R_{j,k,w}$ is the range of the sensor signal within the window w. A stuck response was assigned when the signal range remained below a sensor-specific minimum variation threshold for consecutive windows, as shown in Equation (10):

$$R_{j,k,w} < \theta_j^R \text{for } q\text{consecutive windows} \quad (10)$$

where θ_j^R is the minimum variation threshold for variable (j), and (q) is the required persistence criterion. This condition was used to identify constant-output behavior that remained inside the physically valid measurement range. Abrupt temporal changes were evaluated using the absolute rate of change between consecutive readings. Rate-of-change and local-deviation indicators are commonly used to identify point anomalies, spike events, and transient abnormal behavior in time-series data [16–18]. The rate of change was calculated using Equation (11):

$$ROC_{j,k}(t_i) = \left| \frac{x_{j,k}(t_i) - x_{j,k}(t_i - 1)}{t_i - t_{i-1}} \right| \quad (11)$$

where $ROC_{j,k}(t_i)$ is the absolute rate of change. A spike or transient event was assigned when $ROC_{j,k}(t_i)$ exceeded the sensor-specific threshold under continuous LED-on operation.

Drift-like deviation was evaluated using peer-node residuals rather than the slope of a single sensor signal. This approach was used because gradual changes in temperature, relative humidity, CO_2 concentration, or light intensity may represent real environmental dynamics rather than sensor drift. Residual-based and signal-based approaches are widely used in fault diagnosis to separate abnormal component behavior from normal system variation [16,19]. For each variable, the peer residual of node (k) was calculated using Equation (12):

$$r_{j,k}(t_i) = x_{j,k}(t_i) - median\left(x_{j,\neg k}(t_i)\right) \quad (12)$$

where $r_{j,k}(t_i)$ is the observed peer residual of variable j for node k at timestamp t_i, and $median\left(x_{j,\neg k}(t_i)\right)$ is the median measurement of the same variable from the remaining peer nodes at the same timestamp and operating condition. The peer-node approach did not assume that all cultivation beds had identical environmental conditions. Instead, the contemporaneous median of the remaining nodes was used as a robust spatial reference, and drift-like abnormality was assigned only when both the peer-residual slope and the final residual magnitude exceeded their sensor-specific thresholds for the required persistence period. This combination was intended to reduce sensitivity to small, stable spatial differences among cultivation beds. The residual trend was then estimated using Equation (13):

$$\hat{r}_{j,k,w}(t_i) = \alpha_{j,k,w} + \beta_{j,k,w} t_i \quad (13)$$

where $\hat{r}_{j,k,w}(t_i)$ is the fitted peer-residual trend within window w, $\alpha_{j,k,w}$ is the regression intercept, and $\beta_{j,k,w}$ is the residual trend slope estimated by ordinary least-squares regression of the observed residuals $r_{j,k}(t_i)$ against their actual timestamps within the analysis window. Only records that passed Layer 1 screening were included in the residual-trend regression. Missing timestamp positions, invalid packets, and physically invalid records were excluded rather than interpolated, while the actual server-side timestamps were retained so that irregular sampling intervals were represented by their observed time differences. A drift-like event was assigned only when the residual slope and the final residual magnitude exceeded their sensor-specific thresholds for the required persistence period. This condition reduced the possibility of misclassifying common environmental transitions as sensor drift.

The operating context was used when the required contextual information was available. For light intensity, the continuous LED-on condition was used as the operating context. Persistently low light intensity under this condition was treated as context inconsistency because the measured light signal contradicted the expected continuous lighting condition.

2.8. Proposed Two-Layer Framework

The proposed framework was designed to separate direct signal-quality errors from time-dependent sensor abnormalities. The framework consisted of two sequential layers. Layer 1 performed signal validation using packet status, timestamp continuity, null-value checking, and physical range checking. Layer 2 performed temporal analysis only for records that passed Layer 1. This sequence was used because corrupted, missing, or physically invalid records should not be included in moving-window temporal calculations.

Layer 1 was defined as the signal validation layer. This layer checked missing updates, communication timeouts, null values, non-numeric values, and out-of-range measurements. A communication timeout was identified when the time gap between two consecutive records exceeded the predefined tolerance. An out-of-range measurement was identified when a sensor value exceeded the physical or sensor-specific valid range. Records that failed Layer 1 were assigned immediate Layer 1 labels and were not evaluated by Layer 2. This rule prevented invalid or corrupted records from affecting temporal indicators such as signal range, standard deviation, rate of change, and residual trend slope.

Layer 2 was defined as the temporal analysis layer. This layer evaluated moving-window variability, rate of change, stuck-signal behavior, peer-residual drift-like deviation, and context-based consistency when available. A spike or transient response was associated with an unusually high rate of change over a short interval. Noise or instability was associated with excessive short-term standard deviation or signal range. A stuck response was associated with near-zero variation over consecutive windows. Drift-like deviation was evaluated from the peer-node residual trend rather than from the slope of a single sensor signal. Context-based inconsistency was evaluated when additional operating information

was available. In this study, the continuous LED-on condition was used to evaluate whether the light intensity signal was consistent with the expected lighting condition.

The decision order was fixed before evaluation to make the framework reproducible. Therefore, the assigned primary F-code represented the highest-priority triggered condition according to the predefined hierarchy and did not imply that other abnormal conditions were absent from the same record. For each received record, one primary predicted F-code was assigned using a predefined hierarchy. F1 was assigned first when a packet was missing, delayed, invalid, null, or non-numeric. F2 was assigned when a received value violated the physical or sensor-specific range. Records that passed F1 and F2 were then evaluated by Layer 2. Within Layer 2, the F3 stuck response was evaluated first, followed by F7 operating-context inconsistency, F5 spike or transient event, F4 noise or instability, and F6 peer-residual drift-like deviation. If no rule was triggered, the record was assigned F0 normal operation.

This hierarchy was used to avoid assigning multiple primary fault labels to the same record. For example, a null CO_2 value was assigned as F1 and was not simultaneously evaluated as a temporal fault. Similarly, a physically impossible relative-humidity value above 100% was assigned as F2 and was not included in Layer 2 feature calculation. When more than one Layer 2 condition occurred in the same valid record, the first condition in the predefined hierarchy was assigned as the primary F-code, whereas additional conditions were stored only as secondary diagnostic tags.

The output of the framework was an operating-status label for each sensor variable and node. The normal condition was coded as F0, whereas abnormal or fault-related conditions were coded from F1 to F7 as primary evaluated classes. A study-specific F-code taxonomy was defined to convert detected signal abnormalities into interpretable operating status. F8 was retained only as a secondary suspected node-instability tag because direct voltage and current measurements were not available in this study. Therefore, F8 was not used as a primary evaluated class in the received-record performance analysis. The proposed two-layer framework is shown in Figure 6.

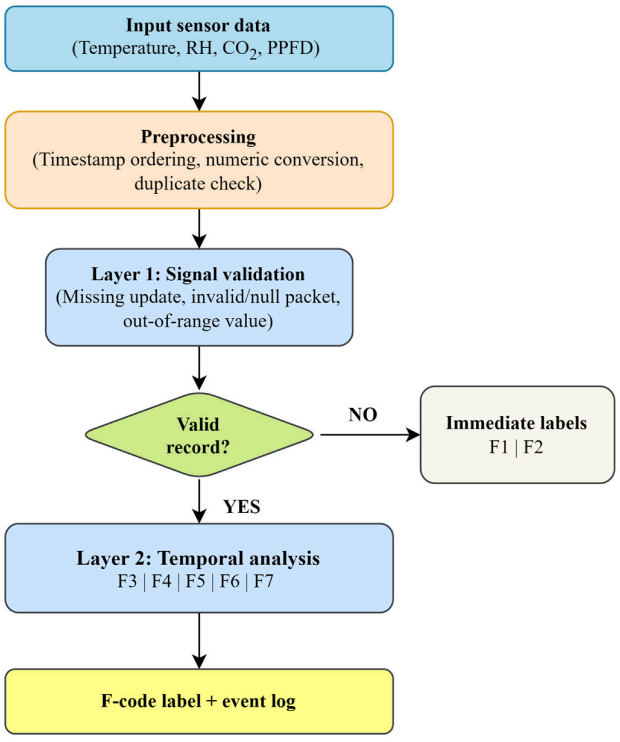


Figure 6. Simplified workflow of the proposed two-layer framework.

The framework first evaluates communication continuity, parsing validity, null-field status, and physical range validity in Layer 1. Measurements that pass Layer 1 are then evaluated using moving-window temporal features, including variability, rate of change, stuck-signal behavior, peer-residual trend, and context-based consistency in Layer 2. The final output is assigned as F0 for normal operation or F1–F7 for primary sensor abnormality or operating-state labels. F8 is retained only as a secondary suspected node-instability tag when multiple simultaneous abnormalities suggest possible power or node-level instability.

2.9. Sensor Operating-Status and Fault-Code Definitions

Sensor operating statuses were defined according to communication status, packet validity, physical validity, temporal signal behavior, and available operating context. Similar data-quality dimensions, including completeness, validity, accuracy, and timeliness, have been widely used in sensor data-quality assessment and IoT data-quality management studies [12,13,22]. In addition, temporal abnormal patterns such as spikes, excessive variability, constant-output behavior, and gradual drift are commonly considered in anomaly detection and sensor fault-diagnosis studies [17,19]. Therefore, the operating status in this study was defined to distinguish communication-related data loss, received-but-invalid records, physically invalid readings, in-range temporal abnormalities, and context-dependent inconsistencies in the vertical farm monitoring system.

A study-specific F-code taxonomy was defined to convert detected signal abnormalities into interpretable operating status. F0 represented normal operation, whereas F1–F7 represented primary abnormal operating status used in the received-record classification. F1 and F2 were assigned from Layer 1 signal validation, while F3–F7 were assigned from Layer 2 temporal or contextual analysis. Each received record was assigned one primary F-code according to the predefined hierarchy described in Section 2.8. This rule was used to avoid assigning multiple primary fault labels to the same record.

F8 was retained only as a secondary suspected multi-channel anomaly tag. Direct voltage and current measurements were not collected in this study; therefore, power or supply failure could not be confirmed from the available dataset. F8 was used only when several sensor channels from the same node showed simultaneous abnormal behavior, repeated invalid packets, or sudden interruption. Thus, F8 was interpreted as a maintenance-support flag for possible node-level instability, not as a primary evaluated fault class. The F-code categories were therefore interpreted as detected abnormal signal or operating-status patterns and not as independently confirmed physical fault causes. The definitions, detection evidence, possible causes, and suggested maintenance actions for the F-code categories are summarized in Table 6.

Table 6. Environmental sensor operating-status and fault-code definitions.

Code	Operating Status	Definition	Main Evidence	Possible Cause	Suggested Action	References
F0	Normal	Data are available, physically valid, and temporally consistent	No validation or temporal rule triggered	Stable sensor operation	Continue monitoring	[12,13,22]
F1	Communication failure	Expected data packet is missing, delayed, corrupted, or repeatedly invalid	Timestamp gap, timeout, absent packet, invalid payload	Wi-Fi dropout, UART/I^2C issue, node reset, broker/server issue	Check network, broker, node reset, wiring	[10,13,24,25]

Table 6. *Cont.*

Code	Operating Status	Definition	Main Evidence	Possible Cause	Suggested Action	References
F2	Out-of-range/invalid value	Measurement exceeds physical or sensor-specific validity limits	$x < L_j$ or $x > U_j$; $RH > 100\%$; negative light or CO_2	Sensor damage, saturation, parsing error	Verify parsing, sensor range, sensor hardware	[12,13,24]
F3	Stuck response	Sensor output remains nearly constant for an abnormal duration	Very small range or $\Delta x \approx 0$ for consecutive windows	Frozen sensor, fouling, software issue, repeated value	Restart/reconnect sensor; check repeated-value buffering	[16,24,25]
F4	Noise/instability	Short-term random fluctuation exceeds the normal baseline level	High SD, high MAD, or high signal range	Electrical noise, unstable supply, local disturbance	Inspect EMI, supply stability, connector	[12,24,25]
F5	Spike/transient	Sudden short-duration jump occurs in the signal	High rate of change or isolated abnormal peak	Transient noise, packet error, abrupt disturbance	Verify transient against neighboring sensors	[17,18,24,25]
F6	Peer-residual drift-like deviation	One node gradually separates from the peer-node baseline	Persistent peer-residual slope and final residual magnitude	Calibration loss, sensor aging, contamination, gradual bias	Recalibrate and compare with peer/reference measurement	[16,19]
F7	Context/cross-sensor inconsistency	Sensor value conflicts with expected operating context or neighboring-node behavior	LED-on but low light, inconsistent neighboring node pattern, impossible joint behavior	Sensor blockage, displacement, local shading, sensor fault, or LED/actuator malfunction	Inspect sensor placement, LED status, and neighboring-node readings before confirming source	[16–19]
F8	Suspected multi-channel node anomaly	Several channels from the same node show simultaneous abnormal behavior or repeated invalid records	Multiple channels abnormal, repeated invalid packets, sudden node-level interruption	Possible power instability, grounding issue, controller reset, communication instability, or connector problem	Check node power, grounding, connector stability, and reset history	[24,25]

2.10. Threshold and Rule Determination

Thresholds were determined before evaluation using sensor specifications, physical limits, reference-normal records from a two-day threshold development period, LED operating context, and peer-node residual behavior. The threshold-development data were collected immediately before the 10-day evaluation period using the same nine-node monitoring system under stable operating conditions. Physical limits were used to identify impossible values, including relative humidity below 0% or above 100%, negative CO_2 concentration, and negative raw light intensity. Sensor specifications were used to define measurement-range limits, resolution-related stuck thresholds, and response-time-related interpretation. Records containing missing updates, invalid packets, physically implausible values, maintenance activities, or documented abnormal operating conditions were excluded from threshold estimation. Thresholds were grouped according to their source. Physical-validity thresholds were obtained from sensor specifications or physically

possible limits, whereas temporal thresholds were estimated from the two-day reference-normal dataset. Baseline-derived temporal thresholds were calculated using either the mean and standard deviation or the robust median and scaled median absolute deviation, depending on the distribution of the corresponding feature. Operating-context thresholds were defined separately from the documented continuous LED-on condition.

For baseline-derived features, the upper threshold was calculated from the reference-normal feature distribution using Equation (14):

$$\theta_j^z = \mu_j^z + \lambda\sigma_j^z \tag{14}$$

where θ_j^z is the threshold for feature z of sensor variable j, μ_j^z is the baseline mean of the feature, σ_j^z is the baseline standard deviation of the feature, and λ is the threshold multiplier. In this study, λ = 3 was fixed before evaluation and was not adjusted using the evaluation dataset. The value of λ = 3 was selected to provide a conservative threshold for identifying deviations from the reference-normal baseline while limiting sensitivity to ordinary short-term variation. For approximately normally distributed features, this setting corresponds to the conventional three-standard-deviation criterion. The three-standard-deviation rule was used as a statistical quality-control criterion for identifying unusual deviations from a stable baseline [29].

For baseline features affected by occasional nonrepresentative spikes, a robust threshold was calculated using the scaled median absolute deviation. The scaled median absolute deviation was calculated using Equation (15):

$$MAD(z_j) = 1.4826 \times median\left(\left|z_j - median(z_j)\right|\right) \tag{15}$$

where $MAD(z_j)$ is the scaled median absolute deviation of feature z for sensor variable j. The factor 1.4826 was used to make MAD comparable to the standard deviation under a normal distribution assumption. This robust scale estimate is less sensitive to outliers than the standard deviation and is suitable when baseline data contain occasional nonrepresentative spikes [30].

The robust threshold was then calculated using Equation (16):

$$\theta_j^z = median(z_j) + \lambda \cdot MAD(z_j) \tag{16}$$

where $(\lambda = 3)$ was fixed before evaluation. Equation 16 was used for features where the reference-normal baseline distribution contained occasional short spikes or non-Gaussian variations. Thus, the mean–SD formulation was used for approximately stable baseline feature distributions, whereas the median–MAD formulation was used when occasional extreme values or non-Gaussian behavior made the standard-deviation estimate less robust.

Because the LED system was operated continuously, light-intensity thresholds were determined under the continuous LED-on baseline condition. No LED-off or scheduled LED-transition class was used in this study. Temperature and relative humidity were expected to change gradually under stable indoor operation. CO_2 concentrations were allowed to vary more strongly because they were affected by ventilation, human activity, and indoor air mixing. Light intensity was evaluated under the continuous LED-on condition; therefore, persistently low light intensity was treated as an operating-context inconsistency rather than as a normal lighting transition.

Detection thresholds were fixed before evaluation using a predefined reference-normal baseline subset. The reference-normal subset contained records collected under stable operation after excluding missing packets, invalid packets, out-of-range values, and fault-injected intervals. Upper thresholds for excessive variability and rate of change were

calculated using either the baseline mean and standard deviation or the robust median MAD formulation, whereas lower thresholds were defined for stuck-response detection. The moving-window length, stride, persistence requirement, multiplier (λ), and final numerical threshold for each sensor-variable and fault-code combination are reported in Table 7. The evaluation dataset and its abnormal labels were not used to adjust the final thresholds.

Table 7. Sensor-specific validity and temporal threshold rules.

Fault Code	Rule Category	Evaluation Scale	Detection Feature	Final Threshold	Persistence Rule
F1	Missing update or invalid packet	Timestamp level or record level	Timestamp gap, packet status	ΔT_i > 60 s, or null, non-numeric, invalid packet	1 event or 1 record
F2	Out-of-range or invalid value	Record level	Sensor value	Temperature <−40 or >80 °C; RH <0 or >100%; CO_2 <0 or >5000 ppm; raw light <0 or above sensor limit	1 record
F3	Stuck response	10 min window	Signal range	$R_T \leq 0.1$ °C; $R_{RH} \leq 0.2\%$ RH; $R_{CO_2} \leq 5$ ppm; $R_L \leq 370$ lx during LED-on	3 consecutive windows
F4	Noise or instability	10 min window	Standard deviation	$\sigma_T > 0.45$ °C; $\sigma_{RH} > 2.5\%$ RH; $\sigma_{CO_2} > 80$ ppm; $\sigma_L > 2222$ lx during LED-on	2 consecutive windows
F5	Spike or transient	Record level	Rate of change	$ROC_T > 0.60$ °C min^{-1}; $ROC_{RH} > 6\%$ RH min^{-1}; $ROC_{CO_2} > 350$ ppm min^{-1}; ROC_L > 8889 lx min^{-1} under continuous LED-on operation	1–2 records
F6	Peer-residual drift-like deviation	60 min window	Residual slope and final residual	Temperature $\lvert \beta_r \rvert > 0.015$ °C min^{-1} and $\lvert r_{end} \rvert > 1.0$ °C; RH $\lvert \beta_r \rvert > 0.04\%$ RH min^{-1} and $\lvert r_{end} \rvert > 3\%$ RH; CO_2 $\lvert \beta_r \rvert > 1.5$ ppm min^{-1} and $\lvert r_{end} \rvert > 100$ ppm; Light $\lvert \beta_r \rvert > 59.3$ lx min^{-1} and $\lvert r_{end} \rvert > 3704$ lx during LED-on	2 consecutive windows
F7	Context inconsistency	10 min window	Window mean and continuous LED-on operating condition	Mean light intensity < 2222 lx under continuous LED-on operation	1 window

Note: Physical limits were based on sensor specifications or physically possible values. Temporal thresholds were derived from the two-day reference-normal dataset using mean ± 3 SD or median ± 3 scaled MAD, as appropriate for the feature distribution. Context thresholds were based on the documented continuous LED-on operating condition.

2.11. Fault Labeling, Validation, and Performance Metrics

A two-day threshold-development period was completed immediately before the 10-day evaluation period. At the 30 s sampling interval, the threshold-development period contained 51,840 expected node-level timestamp positions across the nine sensor nodes. Reference-normal records from this period were used exclusively to characterize stable behavior and determine the baseline-derived thresholds. After threshold determination, all thresholds and decision rules were fixed before application to the 10-day evaluation dataset. No records from the 10-day evaluation period were used to estimate or adjust the detector thresholds. Controlled abnormal scenarios were introduced into copied evaluation records according to a predefined injection protocol that specified the node ID, sensor

variable, fault type, start time, end time, duration, and severity. Fault severity was defined independently of the detector thresholds.

The reference labels for the 10-day evaluation dataset were constructed using three sources. First, reference-normal labels were assigned to stable operating periods after excluding missing timestamp positions, invalid packets, physically invalid values, and injected abnormal intervals. These evaluation labels were not used for threshold determination. Second, controlled software-injected faults were inserted into copied normal records using a predefined injection log. The injection log specified the node ID, sensor variable, fault code, start time, end time, injected value or magnitude, severity level, number of affected records, and reference-label source. Third, naturally observed communication abnormalities were labeled from timestamp continuity and packet-status information. The detection algorithm did not access the injection log or evaluation labels during classification. Accordingly, naturally observed communication and packet-quality abnormalities were distinguished from controlled software-injected F2–F7 abnormal patterns, which were used for reproducible evaluation and were not interpreted as independently confirmed physical sensor failures.

Missing timestamp positions were evaluated separately from received-record classification because no sensor packet existed at those expected time points. These positions were used to quantify communication-related F1 events but were not included in the received-record confusion matrix. For received records, each row was assigned one mutually exclusive primary reference label. The mutually exclusive label was used for primary confusion-matrix evaluation; however, additional simultaneously triggered conditions were considered secondary diagnostic information and were not counted as separate primary classes. F0 represented normal operation, F1 represented received invalid or null packets, F2 represented out-of-range or physically invalid values, and F3–F7 represented temporal or context-dependent abnormalities. F8 was retained only as a secondary suspected multi-channel node-instability tag because voltage and current were not directly measured in this study. Therefore, F8 was not included as a primary evaluated class.

Controlled fault injection was used to generate reproducible abnormal scenarios while preserving the temporal structure of the real monitoring dataset. Out-of-range faults were generated by replacing selected normal values with physically invalid values. Stuck-response faults were generated by holding selected sensor values constant for predefined durations. Noise or instability faults were generated using predefined fluctuation amplitudes. Spike or transient faults were generated using fixed one- or two-record perturbation magnitudes. Drift-like deviations were generated using predefined residual-change rates or final bias magnitudes. Light-intensity context inconsistency was generated by reducing the normal continuous LED-on light signal by predefined loss levels. The final detector thresholds were not used to determine the injected fault magnitudes.

Physically invalid injected faults, temporal software-injected faults, naturally observed communication abnormalities, and suspected multi-channel diagnostic tags were not treated as equivalent ground truth. Software-injected faults were used for received-record performance evaluation because their start time, end time, affected variable, node ID, and injected magnitude were known. Naturally observed missing timestamp positions were reported separately as communication events. Suspected multi-channel node anomalies were retained only for maintenance interpretation and were excluded from the primary performance metrics. The reference-label generation methods, injection conditions, and validation sources used for framework evaluation are summarized in Table 8.

Table 8. Reference-label generation and validation sources used for framework evaluation.

Fault Code	Variable/Node	Fault-Generation Method	Physical/Software Injection	Start/End	Severity	Reference Label Source	Number of Events	Records
F0	All variables/all nodes	Stable normal-operation records	Not injected	Stable operating periods	None	Reference normal evaluation labels	-	227,434
F1	All nodes	Missing timestamp positions	Naturally observed	Timestamp gaps	$\Delta T_i > 60$ s	Expected 30 s timestamp grid	5	290
F1	CO_2/node_06	Invalid or null packet	Naturally observed	Packet-status interval	Null CO_2 field	Packet-status field	1	25
F2	Temperature, RH, CO_2, light	Physically invalid values	Software-injected	Predefined injection log	Outside physical or sensor limits	Injection log	4	40
F3	Temperature/RH selected nodes	Constant-value segment	Software-injected	Predefined injection log	No or near-zero variation	Injection log	3	539
F4	Light during LED-on	High-variance signal	Software-injected	Predefined fluctuation amplitude	±1 SD, ±2 SD, ±4 SD of baseline	Injection log	1	240
F5	CO_2 selected nodes	One- or two-record peak	Software-injected	Fixed one- or two-record spike magnitude	+150, +300, +600 ppm	Injection log	16	32
F6	Air temperature/node_02; relative humidity/node_08; CO_2/node_05	Gradual linear peer-residual shift	Software-injected	2–5 July (RH); 5–8 July (CO_2); 6–9 July (temperature)	Final residuals: +5.4% RH, +168 ppm CO_2, and +1.85 °C	Predefined injection log	3	30,240
F7	Light intensity/node_07	Low light intensity during continuous LED-on operation	Software-injected using operating context	Predefined light-intensity loss level under continuous LED-on operation	25%, 50%, 75%, 90% loss of normal LED output	Injection log and continuous LED-on operating condition	1	360
F8	Multi-channel/node_06	Simultaneous abnormal channel behavior	Suspected naturally observed tag	Event interval	Multi-channel abnormality	Maintenance-support flag only	-	69

Framework performance was evaluated using only received node-level records. Each received record was classified as normal or abnormal for binary performance evaluation and was also assigned one primary F-code for operating-status interpretation. Accuracy, precision, recall, F1 score, specificity, and false-alarm rate were calculated from the true-positive, false-positive, true-negative, and false-negative counts using standard classification definitions commonly applied in anomaly-detection studies [17,18].

Detection delay was calculated only for controlled fault events with known start times. Detection delay was defined as the time difference between the reference fault-start time and the first detected abnormal record. Communication reception delay was analyzed separately because reception delay describes data-transmission timing, whereas detection delay describes the response of the fault-detection algorithm.

3. Results

3.1. Calibration Performance of Environmental Sensors

The calibration performance of the environmental sensors was evaluated before applying the signal-validation and temporal-analysis procedures. The calibration curves were prepared by comparing the sensor-measured values with the corresponding reference-measured values. In this analysis, the sensor-measured value was plotted on the *x*-axis and the reference-measured value was plotted on the *y*-axis. Therefore, the fitted linear equations represented correction relationships for converting raw sensor readings into reference-scale values. Since one representative sensor-node configuration was calibrated, the results were interpreted as calibration performance of the selected sensing configuration rather than independent calibration performance of all nine deployed nodes. Accordingly, the calibration results should not be interpreted as evidence that all nine sensor units had identical calibration responses. The calibration relationships between the sensor-measured and reference-measured values are shown in Figure 7.

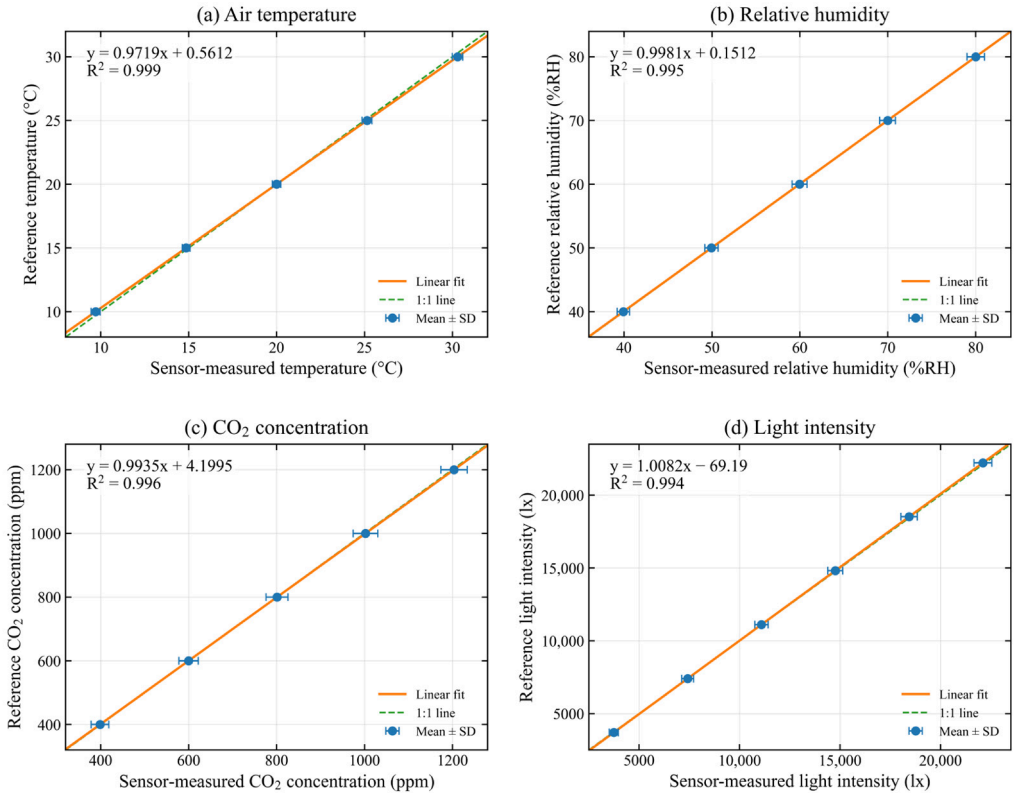


Figure 7. Calibration response of the representative sensor node for (**a**) air temperature, (**b**) relative humidity, (**c**) CO_2 concentration, and (**d**) light intensity. Markers show setpoint means, horizontal error bars show the standard deviation of repeated sensor readings, solid orange lines show the fitted linear regressions, and dashed green lines show the 1:1 relationship. The fitted equations and coefficients of determination (R^2) are reported within each panel.

As shown in Figure 7, all four environmental variables showed strong linear relationships between the sensor-measured and reference-measured values. The coefficient of determination ranged from 0.994 to 0.999, indicating that the selected sensors produced stable and near-linear responses across the tested calibration ranges. The horizontal error bars were small at each calibration condition, showing that the repeated sensor measurements were stable. These results supported the use of the calibrated sensor readings for subsequent preprocessing, data-quality screening, temporal feature extraction, and fault-detection analysis.

For air temperature, the fitted calibration equation was (y = 0.9719x + 0.5612), with an R^2 value of 0.999. The validation MAE, RMSE, and mean bias were 0.20 °C, 0.20 °C, and −0.01 °C, respectively. These results indicated that the calibrated temperature values closely followed the reference values and showed negligible systematic bias. Relative humidity also showed a strong linear response, with the fitted equation (y = 0.9981x + 0.1512) and an R^2 value of 0.995. The validation MAE and RMSE were 1.03% RH and 1.04% RH, respectively, while the mean bias was 0.08% RH. This small positive bias indicated slight overestimation after calibration, but the error remained acceptable for environmental monitoring and temporal fault analysis.

For CO_2 concentration, the fitted calibration equation was y = 0.9935x + 4.1995, with an R^2 value of 0.996. The validation MAE, RMSE, and mean bias were 21.0 ppm, 21.5 ppm, and −1.0 ppm, respectively. Although the absolute error was larger than those of temperature and relative humidity, CO_2 concentration was evaluated over a wider numerical range. The low mean bias indicated limited systematic underestimation after correction. For light intensity, calibration was conducted in lux because both the sensor output and the reference meter measured illuminance. The fitted equation was y = 1.0082x − 69.19, with an R^2 value of 0.994. The validation MAE, RMSE, and mean bias were 496 lx, 504 lx, and 119 lx, respectively. The fitted calibration equations and validation error metrics are summarized in Table 9.

Table 9. Calibration equations and validation error metrics.

Sensor Variable	Calibration Equation	Calibration R^2	Validation MAE	Validation RMSE	Validation Mean Bias
Air temperature	y = 0.9719x + 0.5612	0.999	0.20 °C	0.20 °C	−0.01 °C
Relative humidity	y = 0.9981x + 0.1512	0.995	1.03%RH	1.04%RH	0.08%RH
CO_2 concentration	y = 0.9935x + 4.1995	0.996	21.0 ppm	21.5 ppm	−1.0 ppm
Light intensity	y = 1.0082x−69.19	0.994	496 lx	504 lx	119 lx

3.2. Data Acquisition and Communication Reliability

The data-acquisition performance of the nine-node monitoring system was evaluated using the expected timestamp positions and the received records stored in the local database. Each sensor node was configured to transmit one environmental record every 30 s. Therefore, each node was expected to generate 28,800 records during the 10-day monitoring period, and the complete nine-node system was expected to generate 259,200 node-level records. Because each record contained air temperature, relative humidity, CO_2 concentration, and light intensity, the expected dataset corresponded to 1,036,800 individual sensor measurements.

Across the monitoring period, 258,910 records were received from the expected 259,200 records. This corresponded to an overall data completeness of 99.888% and a missing-data rate of 0.112%. No duplicate node-timestamp records were identified, indicating that the local data-logging structure maintained a consistent time-series format for the nine sensor nodes. The data-acquisition completeness and node-wise missing timestamp positions are shown in Figure 8.

As shown in Figure 8a, the difference between the expected and received records was small compared with the total number of expected records. Figure 8b shows that node-wise completeness ranged from 99.58% to 100.00%. Node_01, node_04, node_06, and node_09 showed complete timestamp coverage, whereas node_03 showed the lowest completeness because 120 timestamp positions were missing. Node_05, node_07, node_02, and node_08 showed 60, 50, 35, and 25 missing timestamp positions, respectively. These results indicate that communication interruptions occurred only occasionally during the monitoring period.

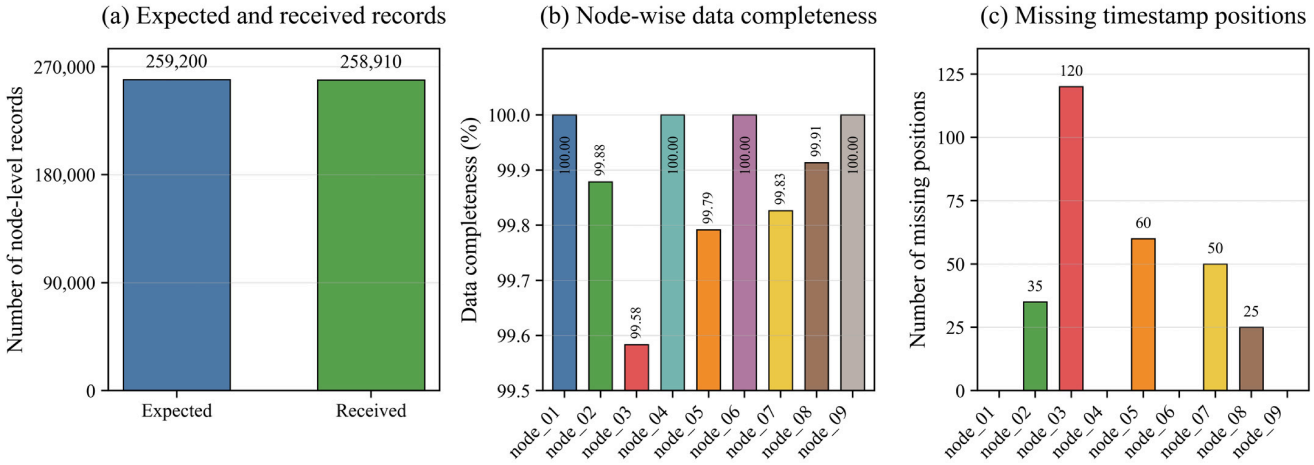


Figure 8. Data-acquisition completeness of the nine-node monitoring system during the 10-day experiment. (**a**) Expected and received node-level records, (**b**) completeness for each node relative to 28,800 expected 30 s timestamp positions, and (**c**) missing timestamp positions by node. In total, 258,910 of 259,200 expected records were received, corresponding to 99.888% completeness and a 0.112% missing-data rate. Missing positions were counted separately from received invalid packets, and no values were interpolated.

Missing timestamp positions were not interpolated or replaced during preprocessing. Instead, they were retained as communication-related missing events and were evaluated separately from received records. This separation was important because missing timestamp positions represented expected packets that were not received, whereas record-level abnormalities were evaluated only among the 258,910 received records. In addition, 25 received records were marked as invalid packets because they contained incomplete sensor fields. These records were not counted as missing timestamp positions because they were received and stored by the local server.

Communication reliability was interpreted using expected record count, received record count, timestamp continuity, invalid-packet status, and missing-update events. Node-to-server reception delay was not reported as an absolute wireless-latency metric because independent synchronization between the sensor-node clocks and the local server clock was not used. Therefore, this section describes data completeness and timestamp regularity rather than absolute communication latency. Overall, the high completeness rate confirmed that the local Wi-Fi to InfluxDB pipeline provided a stable data-acquisition basis for subsequent signal validation and temporal fault analysis.

3.3. Environmental Signal Characteristics Under Normal Operation

The environmental signal characteristics were evaluated using the reference-normal subset of the dataset. Missing timestamp positions, invalid packets, physically invalid values, and reference-labeled abnormal intervals were excluded before this analysis. The resulting normal-operation subset contained 227,434 valid received records. Because the LED system was operated continuously during the monitoring period, the environmental signals were interpreted under a continuous LED-on condition rather than separate LED-on and LED-off periods. The hourly profiles of the calibrated environmental signals are shown in Figure 9.

Air temperature remained stable during normal operation. The overall mean temperature was 23.57 ± 0.52 °C, with values ranging from 21.94 to 24.95 °C. The mean temperature during the facility-active period was 23.61 ± 0.52 °C, while the mean temperature during the facility-inactive period was 23.54 ± 0.52 °C. This small difference indicated that the

thermal condition of the indoor monitoring environment remained stable across the daily operating cycle.

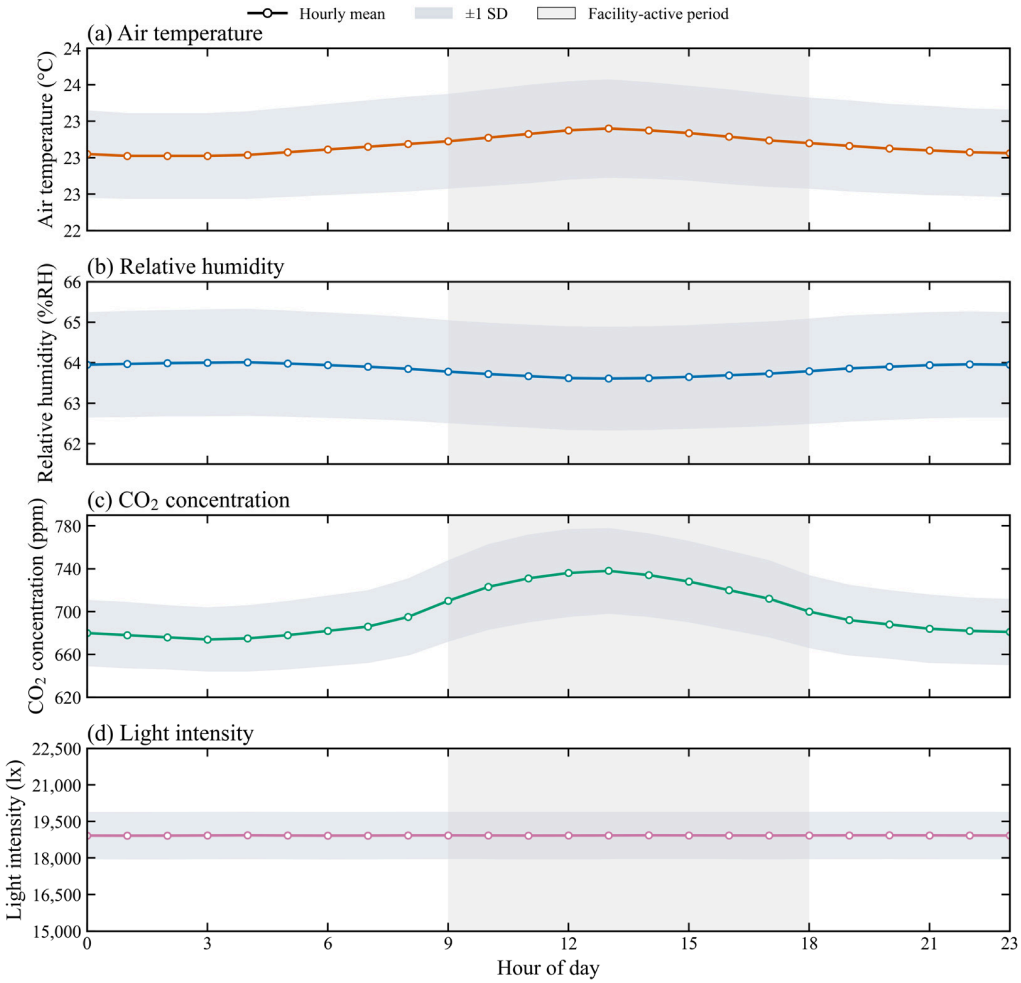


Figure 9. Hourly profiles of calibrated environmental variables in the reference-normal subset (n = 227,434 received records); (**a**) air temperature, (**b**) relative humidity, (**c**) CO_2 concentration, and (**d**) light intensity. Lines and markers show hourly means, blue-gray envelopes show ±1 standard deviation, and the light-gray band marks the facility-active period (09:00–18:00). The LEDs remained on continuously throughout monitoring.

Relative humidity also showed limited variation. The overall mean relative humidity was 63.79 ± 1.33%RH, with values ranging from 59.80 to 68.07%RH. The mean relative humidity was 63.65 ± 1.32%RH during the facility-active period and 63.88 ± 1.33%RH during the facility-inactive period. The slightly lower humidity during the active period was consistent with the small increase in air temperature, but the overall variation remained narrow.

CO_2 concentration showed stronger temporal variations than temperature and relative humidity. The overall mean CO_2 concentration was 698.72 ± 38.81 ppm, with values ranging from 521.98 to 856.06 ppm. The mean CO_2 concentration was higher during the facility-active period, reaching 724.00 ± 34.12 ppm, compared with 683.32 ± 32.99 ppm during the facility-inactive period. Because the experiment was conducted under no-crop conditions, this variation was interpreted as a facility-operation-related pattern, such as human activity, ventilation behavior, or indoor air exchange, rather than crop-related CO_2 uptake.

Light intensity remained stable because the LED system was operated continuously. The overall mean light intensity was 18,921 ± 975 lx, with values ranging from 15,621 to 21,764 lx. The mean light intensity was 18,913 ± 973 lx during the facility-active period and 18,925 ± 976 lx during the facility-inactive period. This confirmed that the light signal remained nearly constant throughout the daily cycle and provided a stable baseline for detecting context-inconsistent light behavior.

Node-wise differences were observed across the nine monitoring nodes. Mean air temperature ranged from 22.71 to 24.18 °C, mean relative humidity ranged from 62.17 to 65.95% RH, and mean CO_2 concentration ranged from 671.95 to 721.88 ppm. These results indicate moderate spatial variability among sensor-node locations and provide node-wise reference patterns for subsequent signal validation and temporal analysis.

3.4. Layer 1 Signal Validation Results

Layer 1 signal validation was applied to identify communication-level and value-level abnormalities before temporal analysis. This layer used timestamp continuity, packet-status checking, missing-value checking, and physical range validation. Unlike Layer 2, which evaluated temporal patterns over moving windows, Layer 1 focused on abnormalities that could be detected directly from expected sampling continuity or from individual received records.

Timestamp continuity analysis detected 290 missing timestamp positions across the 10-day monitoring period. These positions were identified by comparing the expected 30 s sampling interval with the actual timestamp sequence for each node. The missing positions were associated with node_03, node_05, node_07, node_02, and node_08. Node_03 showed the highest missing count, with 120 missing timestamp positions, followed by node_05 with 60, node_07 with 50, node_02 with 35, and node_08 with 25. These missing positions were not interpolated or replaced but were retained as communication-related missing events.

In addition to missing timestamp positions, 25 received records were marked as invalid packets because they contained a null CO_2 field. All invalid packets were associated with node_06. These records were not counted as missing timestamp positions because they were received by the local server. Instead, they were treated as received-but-invalid records and assigned to the Layer 1 validation result.

Physical range validation detected 40 out-of-range received records. These consisted of four air-temperature records, 21 relative-humidity records greater than 100% RH, seven negative CO_2 records, and eight negative raw-light records. These values were physically unrealistic or outside the defined sensor-validity range and were therefore classified as value-level abnormalities. Because the LED system was operated continuously, small negative calibrated light intensity values caused by calibration offset were not treated as faults when the corresponding raw light measurement remained valid. Only clearly negative raw-light values were counted as invalid light records.

Overall, Layer 1 detected 355 abnormal observations, consisting of 290 missing timestamp positions and 65 received abnormal records. The 65 received abnormal records included 25 invalid packets with null CO_2 values and 40 physically out-of-range values. As shown in Figure 10, missing timestamp positions were the dominant Layer 1 abnormality, whereas invalid packets and out-of-range values represented smaller but important signal-quality problems. These results confirmed that Layer 1 provided a necessary screening step before Layer 2 temporal analysis by separating communication failures, incomplete packets, and physically invalid values from valid time-series records.

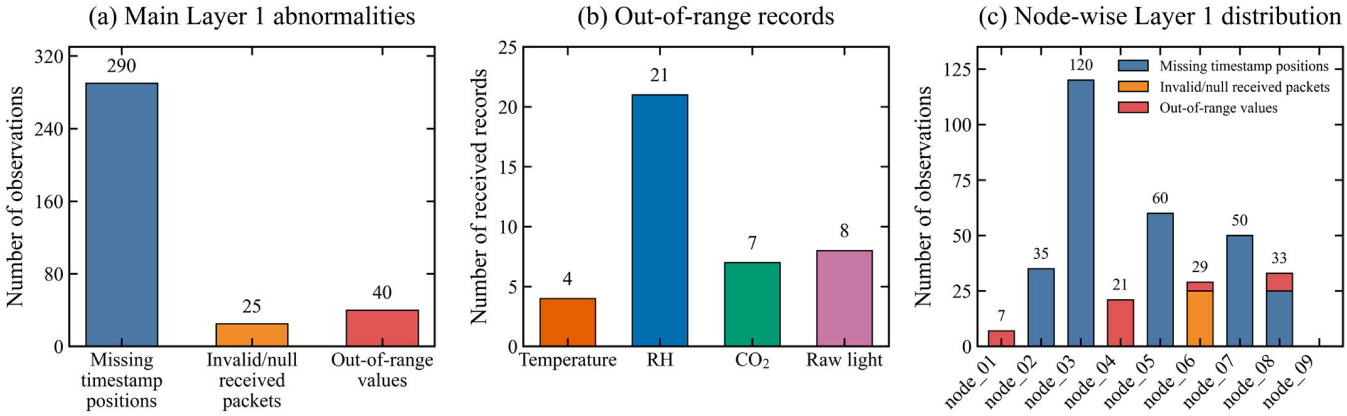


Figure 10. Layer 1 screening results for communication and physical-validity abnormalities. (**a**) Counts of missing timestamp positions, invalid/null received packets, and out-of-range received values; (**b**) out-of-range records by measured variable; and (**c**) node-wise stacked counts of the three Layer 1 categories. Missing timestamp positions were counted against the expected timestamp grid rather than as received records. The 25 null CO_2 packets were assigned once to the invalid-packet category.

3.5. Temporal Analysis Results from Layer 2

After Layer 1 signal validation, the remaining valid records were evaluated using Layer 2 temporal analysis. Layer 2 focused on abnormalities that remained within the physically valid sensor range but showed abnormal temporal behavior. These abnormalities could not be identified reliably using only missing-value checking or physical range validation. Therefore, moving-window range, moving-window standard deviation, rate of change, peer-residual deviation, and operating-context information were used to identify stuck responses, noise or instability, spike or transient events, drift-like deviations, and light-intensity context inconsistency.

The reference-labeled Layer 2 abnormal subset contained 31,411 received records. Stuck-response behavior was assigned to F3 when a signal remained constant or nearly constant over consecutive moving windows. This category included 539 reference-labeled records, and the framework predicted 590 records as F3. Noise or instability was assigned to F4 when short-term signal fluctuation exceeded the adopted threshold. This category included 240 reference-labeled records, while 381 records were predicted as F4. Spike or transient behavior was assigned to F5 when a short-duration abrupt change exceeded the rate-of-change threshold. This category included 32 reference-labeled records, and 181 records were predicted as F5.

Drift-like behavior was assigned to F6 when a node showed persistent peer-residual deviation over the defined temporal window while the measured values remained within the physically valid range. The term “drift-like” was used because the evaluated patterns represented controlled temporal deviations and were not independently confirmed as long-term physical sensor drift. This class contained 30,240 reference-labeled records, and 30,051 records were predicted as F6. The large number of F6 records reflected the long duration of drift-like abnormal intervals rather than frequent short-term events. Therefore, event count and affected-record count were interpreted separately. In contrast, spike events affected only a small number of records because each spike was short.

Context inconsistency was assigned to F7 when light intensity contradicted the expected operating condition. Because the LED system was operated continuously during the experiment, a persistent low-light intensity signal during the LED-on condition was treated as a context-dependent abnormality. This category included 360 reference-labeled records, and 334 records were predicted as F7. Since the light intensity values were numerically

valid, these records were not classified as physical range violations. Instead, they required operating-context interpretation.

Representative Layer 2 temporal patterns are shown in Figure 11. The examples demonstrate why temporal analysis was required after Layer 1 screening. Stuck responses, unstable fluctuations, spike events, drift-like peer-residual deviations, and light intensity context inconsistencies were all possible even when the values remained within the physical sensor range. Therefore, Layer 2 provided the temporal and contextual evidence needed to separate normal environmental variation from abnormal sensor or node behavior.

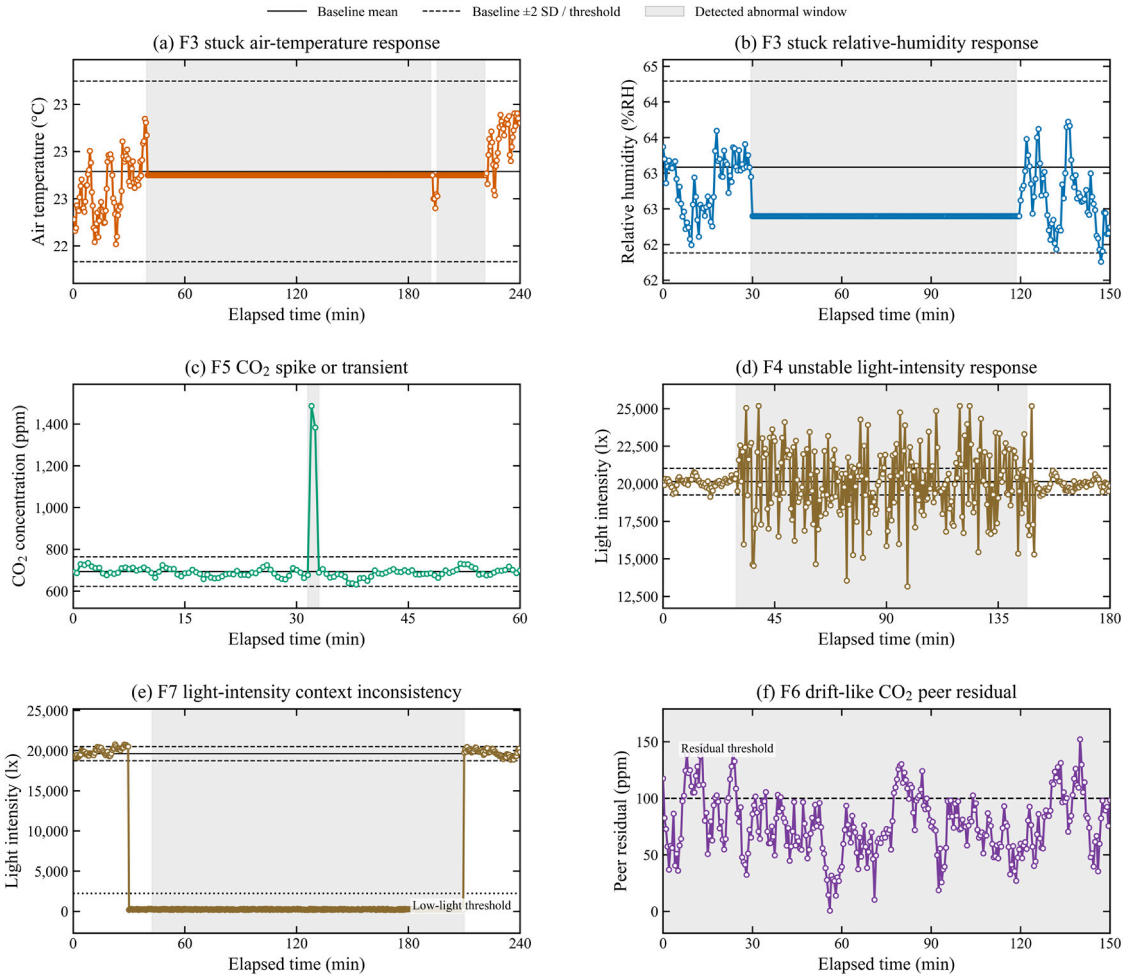


Figure 11. Representative Layer 2 patterns: (**a**) F3 stuck air-temperature response, (**b**) F3 stuck relative-humidity response, (**c**) F5 CO_2 spike/transient, (**d**) F4 unstable light-intensity response, (**e**) F7 low-light context inconsistency during continuous LED-on operation, and (**f**) F6 drift-like CO_2 peer-residual deviation. Gray bands mark detected abnormal windows. Solid black lines show baseline means; dashed lines show baseline ±2 SD or decision thresholds; dotted lines in panel (**e**) show the low-light threshold.

3.6. Overall F-Code Fault Classification and Sensor Operating-Status Summary

The outputs from Layer 1 signal validation and Layer 2 temporal analysis were integrated into the F-code-based operating-status summary. This step converted communication-level, physical-validity, temporal, and context-dependent abnormalities into interpretable operating-status codes. One primary F-code was assigned to each received record according to the predefined decision hierarchy. Missing timestamp positions were summarized separately because no received packet existed at those expected time points.

The received dataset contained 258,910 node-level records. Among the received records, 227,434 records were reference-labeled as F0 normal operation, whereas 227,308 records were predicted as F0 by the proposed framework. The remaining 31,476 received records were reference-labeled as abnormal primary F-codes from F1 to F7. In addition, 290 missing timestamp positions were included in the communication-abnormality summary as F1 observations but were excluded from the received-record classification denominator.

Communication-related abnormalities were assigned to F1. This category included 290 missing timestamp positions and 25 invalid received packets, resulting in 315 F1 observations. Out-of-range and physically invalid measurements were assigned to F2 and included 40 received records. Temporal and context-dependent abnormalities detected by Layer 2 were assigned to F3–F7. These included 539 F3 stuck-response records, 240 F4 noise or instability records, 32 F5 spike or transient records, 30,240 F6 drift-like records, and 360 F7 context-inconsistency records. The F1 missing timestamp positions and invalid/null packets were naturally observed during system operation, whereas the F2–F7 reference abnormalities used for controlled performance evaluation were predominantly software-injected. Therefore, the F2–F7 results were interpreted as detection of predefined abnormal signal patterns rather than independently confirmed natural sensor failures.

The F-code summary showed that event count and affected-record count represented different information. F5 spike or transient behavior occurred in 16 events but affected only 32 records because each spike was short. In contrast, F6 drift-like behavior occurred in three events but affected 30,240 records because the abnormal intervals persisted for a long duration. Therefore, both event-level and record-level summaries were needed to interpret the operating-status results.

F8 was retained only as a secondary suspected multi-channel node-instability tag. Direct voltage and current were not measured in this study; therefore, F8 was not interpreted as a confirmed power-supply or electrical fault. Instead, 69 supporting records from node_06 were flagged as possible node-level instability because multiple abnormal indicators occurred together. These F8 records were used only for maintenance interpretation and were not included as a primary evaluated class.

The F-code-based operating-status summary is shown in Figure 12. The figure separates the overall dataset composition, event-level abnormality counts, and affected observations by F-code. This separation provides a clearer interpretation of short-duration abnormalities, long-duration abnormal intervals, and secondary maintenance-support tags.

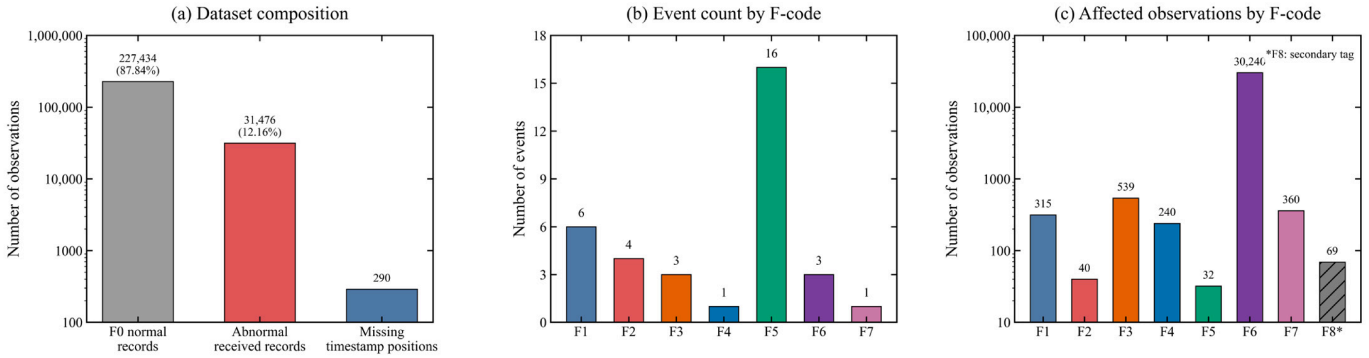


Figure 12. F-code operating-status summary for the 10-day evaluation period. (**a**) Composition of the received-record dataset and missing timestamp positions; (**b**) number of independent events for each primary abnormal F-code; and (**c**) number of affected observations by F-code. F1 combines 290 missing timestamp positions and 25 invalid/null received packets. F2–F7 are primary classes among received abnormal records. F8 * is a secondary maintenance support tag and is excluded from primary class evaluation.

The detailed interpretation of each F-code category is summarized in Table 10. This table clarifies the detection basis, event count, affected observations, and main affected nodes for each operating-status category.

Table 10. Summary of F-code-based operating-status classification.

F-Code	Operating Status	Reference Events	Detected Events	Reference Observations	Detected Observations	Main Affected Node (s)
F0	Normal operation	-	-	227,434 received records	227,308 received records	All nodes
F1a	Invalid/null received packet	1	1	25 received records	25 received records	node_06
F1b	Missing timestamp positions	5	5	290 missing positions	290 missing positions	node_02, node_03, node_05, node_07, node_08
F2	Out-of-range or invalid value	4	4	40 received records	40 received records	node_01, node_04, node_06, node_08
F3	Stuck response	3	3	539 received records	590 received records	node_02, node_05
F4	Noise or instability	1	1	240 received records	381 received records	node_09
F5	Spike or transient	16	15	32 received records	181 received records	node_01, node_03, node_04
F6	Peer-residual drift-like deviation	3	3	30,240 received records	30,051 received records	node_02, node_05, node_08
F7	Context inconsistency	1	1	360 received records	334 received records	node_07
F8	Suspected multi-channel node anomaly	1	1	69 received records	69 received records	node_06

3.7. Framework Performance Evaluation

The proposed two-layer framework was evaluated using received-record reference labels. Missing timestamp positions were not included in the received-record confusion matrix because no sensor packet existed at those expected time points. Similarly, F8 supporting records were not treated as a primary evaluated class because F8 represented a secondary maintenance-support tag rather than an independently confirmed primary fault category. Therefore, the binary performance evaluation was conducted using 258,910 received records, consisting of 227,434 reference-normal records and 31,476 reference-abnormal records.

The binary confusion matrix showed that the framework correctly detected 31,126 abnormal records and correctly retained 226,958 normal records. A total of 476 normal records were incorrectly flagged as abnormal, whereas 350 abnormal records were missed. Based on these values, the proposed framework achieved an accuracy of 99.68%, precision

of 98.49%, recall of 98.89%, F1 score of 98.69%, specificity of 99.79%, and false-alarm rate of 0.21%. These results indicated that the framework detected most abnormal records while preserving most normal records. The binary confusion matrix, overall detection metrics, and F-code-level performance are shown in Figure 13.

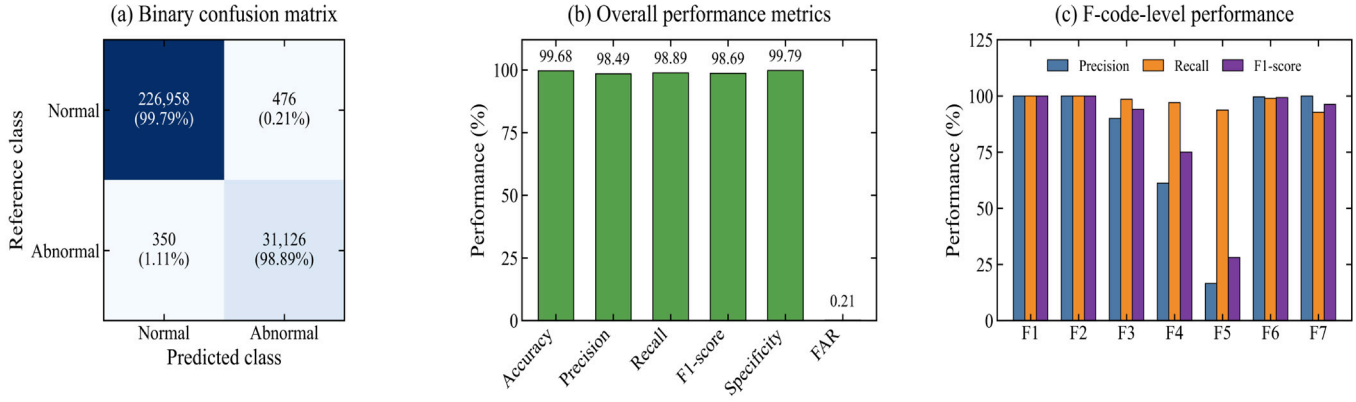


Figure 13. Performance of the two-layer framework on received records from the 10-day evaluation period. (**a**) Binary confusion matrix with counts and row-normalized percentages for reference normal and abnormal records; (**b**) accuracy, precision, recall, F1 score, specificity, and false-alarm rate (FAR); and (**c**) per-class precision, recall, and F1 score for F1–F7. Thresholds were determined from the preceding two-day period and applied without adjustment. Missing timestamp positions and secondary F8 tags were excluded from the received-record confusion matrix.

The F-code-level performance showed that F1 invalid/null received packets and F2 out-of-range values were detected with 100% precision and recall because they were identified using direct packet-validity and physical-range rules. The temporal abnormality classes showed more variable performance. F3 stuck response achieved 90.00% precision, 98.52% recall, and 94.07% F1 score. F4 noise or instability showed lower precision of 61.15% but high recall of 97.08%, indicating that the framework detected most unstable-signal records but also produced additional false-positive detections. F5 spike or transient behavior showed the lowest precision at 16.57%, while recall remained high at 93.75%. Among the 181 records predicted as F5, 30 were true-positive, and 151 were false-positive records. The false-positive detections mainly reflected short-duration CO_2 changes that exceeded the rate-of-change threshold but were not labeled as injected F5 events, including abrupt normal fluctuations and records adjacent to other abnormal intervals. Thus, the adopted rate-of-change rule was sensitive to short transient changes but showed limited specificity for distinguishing injected F5 events from other rapid signal variations.

F6 peer-residual drift-like deviation showed the strongest temporal-class performance, with 99.61% precision, 98.98% recall, and 99.29% F1 score. This result indicated that persistent peer-residual analysis was effective for identifying long-duration drift-like abnormality. F7 context inconsistency achieved 100% precision, 92.78% recall, and 96.25% F1 score. This confirmed that operating-context information helped identify low-light intensity abnormality under continuous LED-on operation while avoiding false-positive classification when the context rule was not triggered.

Detection latency differed among abnormality types. Direct Layer 1 abnormalities, including F1 invalid/null packets and F2 out-of-range values, were detected without delay because they were identified at the received-record level. F5 spike or transient events had a short median alarm latency of 30 s. In contrast, F3 stuck response, F4 noise or instability, F7 context inconsistency, and F6 drift-like deviation required temporal confirmation, resulting in median alarm latencies of 1830 s, 1230 s, 630 s, and 6600 s, respectively. The longest

latency occurred for F6 because gradual drift-like behavior required accumulation of evidence over the peer-residual analysis window before an alarm was assigned.

The detailed received-record performance by F-code is summarized in Table 11. These results showed that the proposed framework was effective for both direct signal-quality abnormalities and longer temporal abnormalities, although short spike-like events and unstable-noise patterns remained more difficult to isolate precisely.

Table 11. F-code-level received-record detection performance during the 10-day evaluation period.

F-Code	Operating Status	Reference Records	TP	FP	FN	Precision (%)	Recall (%)	F1 Score (%)	Events Detected	Median Alarm Latency (s)
F1	Invalid/null received packet	25	25	0	0	100.00	100.00	100.00	1/1	0
F2	Out-of-range or invalid value	40	40	0	0	100.00	100.00	100.00	4/4	0
F3	Stuck response	539	531	59	8	90.00	98.52	94.07	3/3	1830
F4	Noise or instability	240	233	148	7	61.15	97.08	75.04	1/1	1230
F5	Spike or transient	32	30	151	2	16.57	93.75	28.16	15/16	30
F6	Peer-residual drift-like deviation	30,240	29,933	118	307	99.61	98.98	99.29	3/3	6600
F7	Context inconsistency	360	334	0	26	100.00	92.78	96.25	1/1	630

Because the abnormal received-record dataset was dominated by long-duration F6 drift-like records, the binary F1 score mainly represented normal-versus-abnormal screening performance. Therefore, class-level performance was also interpreted using the macro-averaged F-code F1 score. Based on the F1 scores for F1–F7, the macro-averaged F-code F1 score was 84.69%, which was lower than the binary F1 score of 98.69%. This difference reflected the strong class imbalance in the evaluation dataset and indicated that the high binary F1 score should not be interpreted as uniformly high performance across all fault types. Greater weight was therefore given to the class-specific precision, recall, and F1-score results, particularly for the lower-performing F4 and F5 classes.

3.8. Ablation Analysis, Threshold Robustness, and Computational Feasibility

An ablation analysis was conducted to evaluate the contribution of each component of the proposed framework. Four detection configurations were compared: physical/range validation only, temporal rules only without Layer 1 screening, Layer 1 plus Layer 2 without operating-context information, and the full proposed framework with Layer 1, Layer 2, and operating-context analysis. All configurations were assessed using the same 10-day received-record evaluation dataset described in Section 3.7 and the thresholds determined from the preceding two-day period.

Physical/range validation only detected invalid packets and out-of-range values, resulting in 65 true-positive records and no false-positive records. This configuration achieved 100.00% precision but only 0.21% recall and 0.42% F1 score because in-range temporal abnormalities such as stuck responses, noise, spikes, drift-like deviations, and context inconsistencies were not detected. This result confirmed that physical range validation alone was insufficient for sensor fault detection in the vertical farm monitoring dataset.

Temporal rules only improved recall to 98.55% and achieved an F1 score of 96.82%. However, this configuration generated 1581 false-positive records and a false-alarm rate of 0.70%. This occurred because invalid and out-of-range records were not removed before temporal feature extraction, which increased abnormal fluctuations in the moving-window features. When Layer 1 and Layer 2 were combined without operating-context information, the F1 score increased to 98.28%, and the false-alarm rate decreased to 0.17%.

However, recall decreased to 97.83% because light intensity context inconsistency could not be identified without the continuous LED-on operating condition.

The full proposed framework achieved the most balanced performance, with 99.68% accuracy, 98.49% precision, 98.89% recall, and 98.69% F1 score. Compared with the configuration without operating context, the full framework improved recall by detecting light-intensity context inconsistency under continuous LED-on operation. Although the false-alarm rate slightly increased from 0.17% to 0.21%, the overall F1 score improved from 98.28% to 98.69%. These results showed that Layer 1 screening, temporal analysis, and operating-context interpretation were all necessary components of the proposed framework.

Threshold robustness for the short-window temporal rules was evaluated using window lengths of 5, 10, 15, and 20 min while retaining the 60 min peer-residual window for F6. The threshold multiplier was varied from 2.0 to 4.0. Across all tested combinations, accuracy ranged from 97.92% to 99.71%, and F1 scores ranged from 92.05% to 98.80%. Lower multipliers increased recall but also increased false-positive detections. For example, a 5 min window with $\lambda = 2.0$ produced 99.04% recall and a 2.23% false-alarm rate. Larger multipliers reduced false alarms but increased missed abnormal records. The adopted short-window setting of 10 min with $\lambda = 3.0$ achieved 99.68% accuracy, 98.49% precision, 98.89% recall, 98.69% F1 score, and a 0.21% false-alarm rate.

The computational feasibility of the framework was evaluated on the Raspberry Pi 4B local server. Processing the 10-day dataset containing 258,910 received records required 196.4 s in total, corresponding to 0.7586 ms per record. The largest processing cost was associated with Layer 2 moving-window feature extraction, which required 95.6 s, followed by data query from InfluxDB at 38.4 s and peer-residual computation at 27.9 s. The complete pipeline achieved a throughput of 1318.3 records s^{-1} with peak CPU usage of 52.7% and peak memory usage of 412 MB. These results indicated that the proposed rule-based framework was computationally feasible for local server-based monitoring and was well below the 30 s sampling interval used in this study.

The ablation, threshold robustness, and computational feasibility results are summarized in Figure 14. Overall, the results confirmed that the full two-layer framework improved detection balance, remained stable across reasonable threshold settings, and could be executed on a Raspberry Pi-based local server without requiring cloud-based computation.

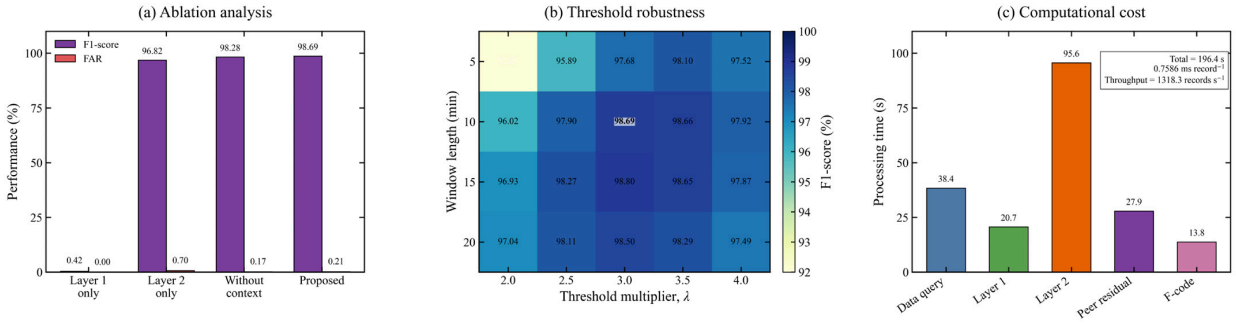


Figure 14. Ablation, threshold-robustness, and computational-cost results for the 10-day evaluation period. (**a**) F1 score and false-alarm rate (FAR) for Layer 1 only, Layer 2 only, the framework without operating context, and the full framework; (**b**) F1 score across 5–20 min short-window lengths and threshold multipliers $\lambda = 2.0$–4.0, with the bold boxed cell marking the adopted 10 min, $\lambda = 3.0$ setting (the F6 peer-residual window remained 60 min); and (**c**) processing time by computational stage on the Raspberry Pi 4B local server.

4. Discussion

The results showed that reliable environmental monitoring in a small-scale vertical farm required more than high data-acquisition completeness. Although the nine-node Wi-Fi monitoring system received 258,910 of 259,200 expected node-level records, corresponding to 99.888% completeness, missing timestamp positions, invalid packets, out-of-range values, and in-range temporal abnormalities were still identified. This confirmed that high data completeness alone does not guarantee sensor-data reliability. Similar data-quality issues, including missingness, invalidity, inconsistency, timeliness errors, and sensor-stream uncertainty, have been reported in IoT and sensor-network studies [10–13,21,22,31–34]. Therefore, environmental monitoring data should be evaluated not only by record availability but also by packet validity, physical plausibility, temporal behavior, and operating context.

The separation between missing timestamp positions and received invalid packets was important for interpretation. Missing timestamp positions represented expected records that were not received, whereas invalid packets represented records that reached the server but contained incomplete sensor fields. These two cases have different maintenance implications. Missing records may indicate temporary Wi-Fi interruption, node reconnection, MQTT transmission disturbance, or local logging interruption, while invalid packets may indicate sensor-interface, parsing, or node-level reading problems. Similar communication and data-stream problems have been reported in wireless sensor networks and IoT monitoring systems, where completeness, timeliness, and validity are key indicators of data quality [10–13,22,24,25,31,32].

The calibration results indicated that the selected sensor configuration provided a suitable measurement baseline after correction. Air temperature, relative humidity, CO_2 concentration, and light intensity showed strong linear relationships between sensor-measured and reference-measured values, with R^2 values ranging from 0.994 to 0.999. However, calibration alone cannot ensure long-term sensor reliability after deployment. Environmental sensors may still show offset, response delay, drift, sensitivity change, or intermittent abnormal readings because of sensor aging, contamination, condensation, dust accumulation, connector instability, or local operating conditions [35–40]. Therefore, calibration should be combined with continuous signal validation and temporal characterization.

The normal-operation analysis provided the baseline for interpreting sensor abnormalities. Under the continuous LED-on operating condition, light intensity remained relatively stable, while air temperature and relative humidity showed narrow variation. CO_2 concentration showed stronger temporal variation, which was attributed to ventilation, indoor air mixing, and human activity because the experiment was conducted under no-crop conditions. This result showed that variable-specific thresholds were necessary. CO_2 required more flexible temporal interpretation, whereas light intensity could be evaluated against the documented continuous lighting condition. Previous vertical-farm and controlled-environment studies have also emphasized that environmental variables differ in their temporal behavior and operational importance [1–4,6–8,41–43].

Layer 1 signal validation effectively detected direct data-quality problems before temporal analysis. Missing timestamp positions, invalid/null packets, and physically out-of-range values were identified before moving-window features were calculated. This step prevented corrupted or physically impossible values from distorting standard deviation, range, rate-of-change, and residual-based indicators. At the same time, these abnormal records were retained as fault-relevant observations rather than being removed silently. This approach is consistent with sensor data-quality studies emphasizing that missingness, invalidity, and inconsistency should be explicitly reported instead of being hidden during preprocessing [10–13,21,22,31–34,39,40].

Layer 2 temporal analysis showed the limitation of static range validation. Stuck responses, unstable signals, spike/transient events, drift-like deviations, and light intensity context inconsistencies occurred within physically valid ranges. Therefore, physical thresholds alone could not detect these abnormalities. Moving-window range and standard deviation were useful for stuck and unstable signals, rate of change was useful for transient spikes, and peer-residual analysis was useful for long-duration drift-like deviations. The F6 class should therefore be interpreted as a drift-like signal deviation rather than confirmed sensor drift, because the evaluated patterns were predominantly software-injected and the 10-day experiment did not provide independent long-term reference measurements for verifying physical sensor drift. However, the peer-node approach did not imply that environmental conditions were identical among the nine cultivation beds. Stable spatial differences were expected and were therefore not treated as abnormalities based on residual magnitude alone; both persistent residual trend and residual magnitude criteria were required. Nevertheless, genuine long-duration spatial changes affecting only one cultivation position could still resemble a drift-like sensor deviation, and this limitation should be considered when interpreting F6 results. These results agree with time-series anomaly-detection and sensor-fault studies showing that point anomalies, contextual anomalies, collective anomalies, stuck-at faults, spikes, noise, and drift require different detection strategies [16–19,23–25,43,44].

The light intensity context inconsistency detected in node_07 demonstrated the value of operating-context information. Because the LED system was operated continuously, persistently low light intensity was numerically possible but inconsistent with the expected lighting condition. This abnormality may indicate sensor blockage, displacement, cable disconnection, local shading, sensor malfunction, or lighting-system abnormality. Accordingly, the detected low-light condition should be interpreted as an operating-context inconsistency rather than confirmation of a specific physical fault. The framework identifies that the measured signal is inconsistent with the documented LED-on condition, but additional inspection or independent measurements are required to determine the actual physical cause. Since artificial lighting strongly affects crop production uniformity, growth response, and energy use in plant factories, light-sensor data should be interpreted together with documented operating conditions [1,4,6–8,42,43]. This result supports the use of contextual anomaly detection rather than relying only on universal physical ranges [17,18].

The F-code taxonomy improved the interpretability of the framework. A binary normal/abnormal result can identify suspicious records, but it does not indicate the likely abnormality type or maintenance response. The proposed F-code structure separates communication failure, out-of-range or invalid values, stuck response, noise or instability, spike or transient behavior, peer-residual drift-like deviation, and context inconsistency. This classification can support practical decisions such as network inspection, sensor-interface checking, recalibration, sensor replacement, or sensor-placement inspection. However, the assigned F-code indicates the detected abnormal signal pattern or operating-state inconsistency and does not independently identify the underlying physical cause. This distinction is particularly important for F7 and F8, where low light, multi-channel abnormality, or communication instability may have several possible causes that were not independently confirmed in this study. Because the fixed hierarchy can suppress lower-priority conditions when multiple rules are triggered simultaneously, the primary F-code should be interpreted as the prioritized operating-status label rather than as evidence that only one abnormal signal characteristic was present. The frequency of simultaneously triggered Layer 2 rules was not separately quantified in the present evaluation; therefore, the extent of multi-rule overlap could not be assessed retrospectively. Future implementations should retain all rule-level trigger states to allow explicit analysis of overlapping abnormal conditions. In-

terpretable operating-status categories are important for practical sensor fault diagnosis because operators need actionable information, not only anomaly scores [16,19,23–25,45].

The performance results confirmed that the two-layer framework achieved balanced detection. The framework obtained 99.68% accuracy, 98.49% precision, 98.89% recall, 98.69% F1 score, and 99.79% specificity for received-record classification. Direct Layer 1 abnormalities were detected with high reliability because they were based on packet-validity and physical-range rules. Temporal classes showed more variable precision because unstable and spike-like signals could overlap with short normal fluctuations or neighboring abnormal windows. In particular, F4 and F5 were more difficult to isolate precisely than long-duration F6 drift-like deviations. The low F5 precision (16.57%) indicated that the rate-of-change criterion was sensitive to rapid signal variations that were not uniquely associated with the injected spike/transient class. Normal abrupt CO_2 fluctuations and records near other abnormal intervals could therefore exceed the F5 threshold and be classified as spike-like events. Although adjustment of the F5 threshold could improve precision, the predefined threshold was not re-optimized using the evaluation dataset to avoid post hoc tuning and information leakage. Further validation should therefore examine F5-specific threshold optimization using an independent development dataset and a larger number of transient events. This result is consistent with previous anomaly-detection studies showing that short-duration and noisy anomalies are often more difficult to classify precisely than persistent abnormal patterns [16–18,23–25].

Although the overall binary abnormality-detection metrics were high, class-specific performance should also be considered because the abnormal dataset was dominated by long-duration F6 drift-like records. In contrast, the short-duration F4 and F5 classes were represented by substantially fewer events and showed lower precision, particularly F5. Therefore, the overall binary F1 score should be interpreted primarily as evidence of effective normal-versus-abnormal screening rather than uniformly high fault-type classification performance. Accordingly, the macro-averaged F-code F1 score of 84.69% was considered a more representative indicator of balanced multiclass performance than the binary F1 score alone. Future studies should include a larger number and broader severity range of short-duration abnormal events and report macro-averaged, event-level, and record-level performance to provide a more balanced assessment of multiclass fault-detection capability.

The ablation and robustness analyses supported the proposed framework design. Physical/range validation alone missed most in-range temporal abnormalities, while temporal analysis alone increased false alarms because invalid records were not removed before feature extraction. Combining Layer 1 and Layer 2 improved the precision–recall balance, and adding operating-context information enabled detection of light intensity inconsistency under continuous LED-on operation. The adopted 10 min moving window and $\lambda = 3.0$ provided a practical balance between sensitivity and false-alarm control. This result is consistent with statistical quality-control and robust anomaly-detection principles, where threshold selection must balance abnormal-event sensitivity and false-alarm reduction [17,18,20,46].

The computational analysis showed that the framework was feasible for local server-based processing. The average processing time per received record was approximately 0.76 ms, substantially shorter than the 30 s sampling interval, indicating adequate computational capacity for sequential online implementation. Therefore, the proposed rule-based approach showed feasibility for near-real-time sensor fault screening under the tested experimental configuration without requiring cloud-based computation or complex machine-learning models. This is useful for small-scale research vertical farms, where interpretability, low computational cost, and local deployment may be more practical than highly model-dependent approaches [16–19,23,26–28,45,47–49]. However, this re-

sult represents processing feasibility, not absolute node-to-server communication latency, because independent clock synchronization between sensor nodes and the server was not documented.

Several limitations remain. First, the experiment covered 10 days and nine sensor nodes in one small-scale vertical farm, so longer-term validation is needed. Second, the empty-bed design did not account for crop-induced changes in CO_2 concentration, relative humidity, air temperature, airflow, or canopy-level light distribution. The presence of plants may also increase spatial and vertical variability among cultivation beds, which could affect the baseline distributions, peer-node residuals, and temporal thresholds used by the proposed framework. Therefore, the present results should be interpreted as validation under the tested empty-bed configuration rather than as direct evidence of equivalent performance under crop-growing conditions. Third, one representative sensor-node configuration was calibrated; therefore, possible sensor-to-sensor variation among the nine deployed units was not independently quantified, and independent calibration evidence was not available for the remaining eight deployed nodes. Future work should include node-specific calibration or independent reference measurements for each deployed sensor unit. Fourth, most temporal and value-level abnormality classes used for performance evaluation were based on controlled software-injected patterns rather than independently confirmed natural sensor failures. Therefore, the reported detection metrics should be interpreted as performance under the defined experimental reference-label conditions rather than comprehensive validation against naturally occurring physical sensor faults. Fifth, although threshold development and final evaluation were chronologically separated, both periods originated from the same sensor deployment and vertical-farm facility. Future validation should therefore assess transferability using independent monitoring periods, sensor units, and facilities. Sixth, the two-day threshold-development period may not have captured the full range of normal long-term environmental variability. Seventh, the small number of independent F3, F4, F6, and F7 events limits event-level generalization even when long events contain many affected records. Finally, F8 was retained only as a secondary suspected multi-channel node-instability tag because voltage and current were not measured.

Overall, the proposed framework provided a transparent and maintenance-oriented approach for environmental sensor data characterization in vertical farms. By integrating communication-quality screening, physical signal validation, temporal feature analysis, peer-node residual comparison, operating-context interpretation, and F-code classification, the framework can help identify sensor abnormalities before the data are used for monitoring or control decisions. The transferability of the current thresholds and peer-node comparisons to crop-growing conditions therefore remains to be evaluated. Future validation under longer-term crop-growing conditions will be necessary to evaluate its robustness and broader transferability in more complex vertical-farm environments.

5. Conclusions

An interpretable two-layer signal validation and temporal analysis framework was developed and evaluated for environmental sensor fault detection in a Wi-Fi-based small-scale vertical farm. Nine sensor nodes were installed across nine cultivation beds, and air temperature, relative humidity, CO_2 concentration, and light intensity were monitored at 30 s intervals. Detection thresholds were determined from a two-day period immediately preceding the evaluation and then applied without adjustment to the subsequent 10-day dataset. The LED system was operated continuously, and the experiment was conducted under empty-bed conditions to focus on sensor-data reliability and fault-detection performance rather than crop-growth response. The framework was therefore validated

under a 10-day, nine-node, single-facility, empty-bed experimental configuration, with most temporal abnormality classes evaluated using controlled software-injected patterns.

The monitoring system received 258,910 records from 259,200 expected node-level records, corresponding to 99.888% data completeness. Layer 1 detected direct signal-quality abnormalities, including 290 missing timestamp positions, 25 invalid/null received packets, and 40 physically out-of-range received records. These records were retained as fault-relevant observations and were not interpolated or removed silently. Layer 2 then evaluated only records that passed Layer 1 screening and detected in-range temporal abnormalities, including stuck responses, noise or instability, spike or transient events, peer-residual drift-like deviations, and light intensity context inconsistency under continuous LED-on operation. The evaluation combined naturally observed communication and packet-quality abnormalities with predominantly controlled software-injected value and temporal abnormal patterns.

The framework achieved 99.68% accuracy, 98.49% precision, 98.89% recall, a 98.69% binary F1 score, and 99.79% specificity for received-record screening. The macro-averaged F-code F1 score was 84.69%, and greater emphasis was placed on this class-balanced metric and the individual F-code results because the evaluation dataset was strongly imbalanced. Overall, normal-versus-abnormal screening was stronger than uniform fault-type classification, and spike/transient identification remained the most challenging temporal class. The F-code taxonomy converted detected abnormalities into operating-status categories that can support sensor maintenance and monitoring decisions.

The ablation and robustness analyses showed that physical validation alone was insufficient, temporal analysis improved detection of in-range abnormalities, and operating-context information was necessary for identifying light intensity inconsistency under continuous LED-on operation. The computational evaluation showed that the rule-based pipeline was feasible for local processing on a Raspberry Pi 4B server under the 30 s sampling interval. However, absolute node-to-server communication latency was not evaluated because independent clock synchronization was not documented.

The main contribution of this study was the development of a transparent and maintenance-oriented framework that integrates direct signal validation, temporal behavior analysis, peer-node residual comparison, operating-context interpretation, and F-code-based fault classification for environmental sensor data characterization. Future work should validate and, where necessary, recalibrate the framework under longer-term crop-growing conditions, where plant effects on environmental dynamics and spatial variability are expected to be greater. It should also include node-specific calibration and independent reference measurements, collect voltage/current information for confirming node-level instability, and test the approach under different lighting, ventilation, and crop-production conditions.

Author Contributions: Conceptualization, M.A.R., M.N.R., and S.-O.C.; methodology, M.A.R., M.N.R. and S.-O.C.; software, M.A.R., M.N.R. and E.Z.; validation, M.A.R., M.N.R., E.Z., J.-Y.K., M.M.R. and S.R.; formal analysis, M.A.R., M.N.R., E.Z., J.-Y.K., M.M.R. and S.R.; investigation, M.N.R., J.-Y.K. and S.-O.C.; resources, S.-O.C.; data curation, M.A.R., M.N.R., E.Z., J.-Y.K., M.M.R. and S.R.; writing—original draft preparation, M.A.R.; writing—review and editing, M.N.R., E.Z., J.-Y.K. and S.-O.C.; visualization, M.A.R., M.M.R. and S.R.; supervision, S.-O.C.; project administration, S.-O.C.; funding acquisition, S.-O.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Korea Institute of Planning and Evaluation for Technology in Food, Agriculture, and Forestry (IPET) through the K-Vertical Farm Globalization Project funded by the Ministry of Agriculture, Food, and Rural Affairs (MAFRA) (Project No. RS-2025-02303373), Republic of Korea.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest. Author Mr. Ji-Yeon Kang was employed by the company Bonghwa Insect Hotel Agricultural Corporation, Ltd., Bonghwa 36218, Republic of Korea. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The authors declare that this study received no commercial funding from any companies and that no companies were involved in this study.

References

1. Kozai, T.; Niu, G.; Takagaki, M. *Plant Factory: An Indoor Vertical Farming System for Efficient Quality Food Production*; Academic Press: London, UK, 2016.
2. Benke, K.; Tomkins, B. Future food-production systems: Vertical farming and controlled-environment agriculture. *Sustain. Sci. Pract. Policy* **2017**, *13*, 13–26. [CrossRef]
3. Vatistas, C.; Avgoustaki, D.D.; Bartzanas, T. A systematic literature review on controlled-environment agriculture: How vertical farms and greenhouses can influence the sustainability and footprint of urban microclimate with local food production. *Atmosphere* **2022**, *13*, 1258. [CrossRef]
4. van Gerrewey, T.; Boon, N.; Geelen, D. Vertical farming: The only way is up? *Agronomy* **2022**, *12*, 2. [CrossRef]
5. Akpenpuun, T.D.; Sanusi, H.O.; Ogundele, O.M.; Usman, A.; Gbenga, A.R.; Ogunlowo, Q.O. Advancements in vertical farming: A review of potentials, challenges, and prospects. *Korean J. Agric. Sci.* **2025**, *52*, 619–638. [CrossRef]
6. Morella, P.; Lambán, M.P.; Royo, J.; Sánchez, J.C. Vertical farming monitoring: How does it work and how much does it cost? *Sensors* **2023**, *23*, 3502. [CrossRef]
7. Saad, M.H.M.; Hamdan, N.M.; Sarker, M.R. State of the art of urban smart vertical farming automation system: Advanced topologies, issues and recommendations. *Electronics* **2021**, *10*, 1422. [CrossRef]
8. Shamshiri, R.R.; Kalantari, F.; Ting, K.C.; Thorp, K.R.; Hameed, I.A.; Weltzien, C.; Ahmad, D.; Shad, Z.M. Advances in greenhouse automation and controlled environment agriculture: A transition to plant factories and urban agriculture. *Int. J. Agric. Biol. Eng.* **2018**, *11*, 1–22. [CrossRef]
9. Kim, Y.-S.; Kim, D.-H.; Chung, S.-O.; Choi, C.-H.; Choi, T.-H.; Kim, Y.-J. Development of an environment field monitoring system to measure crop growth. *Korean J. Agric. Sci.* **2019**, *46*, 57–65. [CrossRef]
10. Karkouch, A.; Mousannif, H.; Al Moatassime, H.; Noel, T. Data quality in Internet of Things: A state-of-the-art survey. *J. Netw. Comput. Appl.* **2016**, *73*, 57–81. [CrossRef]
11. Liu, C.; Nitschke, P.; Williams, S.P.; Zowghi, D. Data quality and the Internet of Things. *Computing* **2020**, *102*, 573–599. [CrossRef]
12. Teh, H.Y.; Kempa-Liehr, A.W.; Wang, K.I.K. Sensor data quality: A systematic review. *J. Big Data* **2020**, *7*, 11. [CrossRef]
13. Zhang, L.; Jeong, D.; Lee, S. Data quality management in the Internet of Things. *Sensors* **2021**, *21*, 5834. [CrossRef]
14. Ni, K.; Ramanathan, N.; Chehade, M.N.H.; Balzano, L.; Nair, S.; Zahedi, S.; Kohler, E.; Pottie, G.; Hansen, M.; Srivastava, M. Sensor network data fault types. *ACM Trans. Sens. Netw.* **2009**, *5*, 25:1–25:29. [CrossRef]
15. Sharma, A.B.; Golubchik, L.; Govindan, R. Sensor faults: Detection methods and prevalence in real-world datasets. *ACM Trans. Sens. Netw.* **2010**, *6*, 23:1–23:39. [CrossRef]
16. Liang, S.; Liu, P.; Zhang, Z.; Wu, Y. Research on fault diagnosis of agricultural IoT sensors based on improved dung beetle optimization–support vector machine. *Sustainability* **2024**, *16*, 10001. [CrossRef]
17. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Comput. Surv.* **2009**, *41*, 15:1–15:58. [CrossRef]
18. Blázquez-García, A.; Conde, A.; Mori, U.; Lozano, J.A. A review on outlier/anomaly detection in time series data. *ACM Comput. Surv.* **2021**, *54*, 1–33. [CrossRef]
19. Zou, X.; Liu, W.; Huo, Z.; Wang, S.; Chen, Z.; Xin, C.; Bai, Y.; Liang, Z.; Gong, Y.; Qian, Y.; et al. Current status and prospects of research on sensor fault diagnosis of agricultural Internet of Things. *Sensors* **2023**, *23*, 2528. [CrossRef]
20. Jihani, N.; Kabbaj, M.; Benbrahim, M. Sensor fault detection and isolation for smart irrigation wireless sensor network based on parity space. *Int. J. Electr. Comput. Eng.* **2023**, *13*, 1463–1471. [CrossRef]
21. Fizza, K.; Jayaraman, P.P.; Banerjee, A.; Georgakopoulos, D.; Ranjan, R. Evaluating sensor data quality in Internet of Things smart agriculture applications. *IEEE Micro* **2021**, *42*, 51–60. [CrossRef]
22. Wang, R.Y.; Strong, D.M. Beyond accuracy: What data quality means to data consumers. *J. Manag. Inf. Syst.* **1996**, *12*, 5–33. [CrossRef]

23. Borrego, C.; Costa, A.M.; Ginja, J.; Amorim, M.; Coutinho, M.; Karatzas, K.; Sioumis, T.; Katsifarakis, N.; Konstantinidis, K.; De Vito, S.; et al. Assessment of air quality microsensors versus reference methods: The EuNetAir joint exercise. *Atmos. Environ.* **2016**, *147*, 246–263. [CrossRef]
24. Shekarian, S.M.; Aminian, M.; Fallah, A.M.; Moghaddam, V.A. AI-powered sensor fault detection for cost-effective smart greenhouses. *Comput. Electron. Agric.* **2024**, *224*, 109198. [CrossRef]
25. Munser, L.; Hoyo, Á.; Petzke, F.; Moreno, J.A.; Streif, S. Residual-based fault detection and isolation in control environment agriculture. *J. Process Control* **2026**, *157*, 103607. [CrossRef]
26. Rahman, M.A.; Reza, M.N.; Bicamumakuba, E.; Samsuzzaman; Jin, H.; Chung, S.-O.; Hong, S.J. ICT-based status monitoring and anomaly detection techniques of smart greenhouse components: A review. *Precis. Agric. Sci. Technol.* **2025**, *7*, 170–188. [CrossRef]
27. Bicamumakuba, E.; Reza, M.N.; Jin, H.; Choi, H.; Chung, S.-O. Mobile application for signal processing and abnormality detection of ambient environmental sensors in a smart greenhouse. *Agronomy* **2026**, *16*, 820. [CrossRef]
28. Choe, H.-O.; Lee, M.-H. Artificial intelligence-based anomaly detection technology for equipment condition monitoring in smart farms. *Appl. Sci.* **2025**, *15*, 12843. [CrossRef]
29. Montgomery, D.C. *Introduction to Statistical Quality Control*, 6th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2009.
30. Rousseeuw, P.J.; Croux, C. Alternatives to the median absolute deviation. *J. Am. Stat. Assoc.* **1993**, *88*, 1273–1283. [CrossRef]
31. Klein, A.; Lehner, W. Representing data quality in sensor data streaming environments. *ACM J. Data Inf. Qual.* **2009**, *1*, 10:1–10:28. [CrossRef]
32. Kümper, D.; Iggena, T.; Tönjes, R.; Pulvermüller, E. Valid.IoT: A framework for sensor data quality analysis and interpolation. In Proceedings of the 9th ACM Multimedia Systems Conference, Amsterdam, The Netherlands, 12–15 June 2018; pp. 294–303. [CrossRef]
33. Zubair, N.; Niranjan, A.; Hebbar, K.; Simmhan, Y.L. Characterizing IoT data and its quality for use. *arXiv* **2019**, arXiv:1906.10497. [CrossRef]
34. Mansouri, T.; Sadeghi Moghadam, M.R.; Monshizadeh, F.; Zareravasan, A. IoT data quality issues and potential solutions: A literature review. *Comput. J.* **2023**, *66*, 615–625. [CrossRef]
35. Maag, B.; Zhou, Z.; Thiele, L. A survey on sensor calibration in air pollution monitoring deployments. *IEEE Internet Things J.* **2018**, *5*, 4857–4870. [CrossRef]
36. Concas, F.; Mineraud, J.; Lagerspetz, E.; Varjonen, S.; Liu, X.; Puolamäki, K.; Nurmi, P.; Tarkoma, S. Low-cost outdoor air quality monitoring and sensor calibration: A survey and critical analysis. *ACM Trans. Sens. Netw.* **2021**, *17*, 1–44. [CrossRef]
37. Okafor, N.U.; Alghorani, Y.; Delaney, D.T. Improving data quality of low-cost IoT sensors in environmental monitoring networks using data fusion and machine learning approach. *ICT Express* **2020**, *6*, 220–228. [CrossRef]
38. Mourad, N. Robust smoothing of one-dimensional data with missing and/or outlier values. *IET Signal Process.* **2021**, *15*, 323–336. [CrossRef]
39. Hayes, M.H. *Statistical Digital Signal Processing and Modeling*; John Wiley & Sons: New York, NY, USA, 1996.
40. Leis, J.W. *Digital Signal Processing Using MATLAB for Students and Researchers*; John Wiley & Sons: Hoboken, NJ, USA, 2011. [CrossRef]
41. Ullah, R.; Asghar, I.; Griffiths, M.G.; Stacey, C.; Stiles, W.; Whitelaw, C. Internet of things-based sensor system for vertical farming and controlled environment agriculture. In Proceedings of the 2023 6th Conference on Cloud and Internet of Things (CIoT), Lisbon, Portugal, 20–22 March 2023; pp. 136–140.
42. Kim, B.-S. Effects of CO_2 Concentration and Greenhouse Internal Temperature on the Production of the Hard-Boiled Tomato 'Dephnis' Variety. Master's Thesis, Graduate School of Sunchon National University, Suncheon, Republic of Korea, 2023.
43. Yang, Y.; Ding, S.; Liu, Y.; Meng, S.; Chi, X.; Ma, R.; Yan, C. Fast wireless sensor for anomaly detection based on data stream in an edge-computing-enabled smart greenhouse. *Digit. Commun. Netw.* **2022**, *8*, 498–507. [CrossRef]
44. Aggarwal, C.C. (Ed.) *Managing and Mining Sensor Data*; Springer: New York, NY, USA, 2013. [CrossRef]
45. Chung, S.-O.; Ulianti, G.A.P.M.; Zakir, E.; Jin, H.; Yun, C.M. Operation status monitoring of sensors and actuators for smart vertical farming. In *Sensing for Agriculture and Food Quality and Safety XVIII*; SPIE: Bellingham, WA, USA, 2026; Volume 14053, p. 140530B. [CrossRef]
46. Park, M.-H.; Chakraborty, S.; Vuong, Q.; Noh, D.-H.; Lee, J.-W.; Lee, J.-U.; Choi, J.-H.; Lee, W.-J. Anomaly detection based on time series data of hydraulic accumulator. *Sensors* **2022**, *22*, 9428. [CrossRef]
47. Torabi, H.; Mirtaheri, S.L.; Greco, S. Practical autoencoder-based anomaly detection by using vector reconstruction error. *Cybersecurity* **2023**, *6*, 1. [CrossRef]

48. Zanella, A.R.A.; da Silva, E.; Albini, L.C.P. CEIFA: A multi-level anomaly detector for smart farming. *Comput. Electron. Agric.* **2022**, *202*, 107279. [CrossRef]
49. Uyeh, D.D.; Akinsoji, A.; Asem-Hiablie, S.; Bassey, B.I.; Osinuga, A.; Mallipeddi, R.; Amaizu, M.; Ha, Y.; Park, T. An online machine learning-based sensors clustering system for efficient and cost-effective environmental monitoring in controlled environment agriculture. *Comput. Electron. Agric.* **2022**, *199*, 107139. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.